Punyashlok Ahilyadevi Holkar Solapur University, Solapur



Name of the Faculty: Science and Technology

NEP-2020

Syllabus: - BCA (Science)

Name of the Program: BCA (Science)-I (Sem.- I & II) (NEP-2020)

(Syllabus to be implemented w.e.f. June 2024)

Punyashlok Ahilyadevi Holkar Solapur University, Solapur

BCA (Bachelor of Computer Application)

Preamble: Bachelor of Computer Applications (BCA) is a four year under Graduate Programme. The Course is designed to bridge the gap between IT industries and Academic institutes by incorporating the latest development, into the Curriculum and to give students a complete understanding within a structured framework. The Course helps the students to build-up a successful Career in Computer Science and for pursuing higher studies in Computer Science.

Objectives of the Programme:

- 1. Demonstrate the ability to adapt to technological changes and innovations in the discipline.
- 2. Develop computer programs using functional programming and object-oriented programming paradigms.
- 3. To train students in professional skills related to Software Industry.
- 4. To prepare the necessary knowledge base for research and development in Computer Science.
- 5. To help students build-up a successful career in Computer Science and to produce entrepreneurs who can innovate and develop software products.
- 6. An ability to apply knowledge of mathematics, statistics and computer science in practice.
- 7. An ability to enhance not only a comprehensive understanding of the theory but its application too in diverse fields.
- 8. The program prepares the young professional for a range of computer applications, computer organization, and techniques of Computer Networking, Software Engineering, Web Development, Database management and Advance Java.
- 9. An ability to design a computing system to meet desired needs within realistic constraints such as safety, security and applicability in multidisciplinary teams with a positive attitude.
- 10. In order to enhance the programming skills of the young IT professionals, the program has introduced the concept of project development in each language/technology learned during the curriculum.

Eligibility for BCA (Bachelor of Computer Application):

The candidate passing the Higher Secondary Examination Conducted by the Maharashtra State Board of Higher Secondary Education with Arts/Commerce/Science stream or its equivalent or any Diploma of not less than two years.

Programme Outcomes (PO):

These outcomes describe what students are expected to know and can do by the time of graduation. They relate to the skills, knowledge, and behavior's that students acquire in their graduation through the program

Programme Outcomes for BCA (Bachelor of Computer Application):

The BCA (Bachelor of Computer Application) programme enables students to attain, by the time of graduation:

PO1: Design and develop software based solutions for real life problems, serving effectively to the requirements of computer field and Society.

PO2: Attain sufficient knowledge related to computer domains, possesses technical, soft and hard skills and apply them effectively in team work.

PO3: Ability to link knowledge of Computer Science with other two chosen auxiliary disciplines of study.

PO4: Display ethical code of conduct in the usage of Internet and Cyber systems.

PO5: Ability to pursue higher studies of specialization and to take up technical employment.

PO6: Identify, formulate and analyze complex real-life problems in order to arrive at computationally viable conclusions using fundamentals of mathematics, computer sciences, management and relevant domain disciplines.

PO7: Ability to operate, manage, deploy, configure computer network, hardware, and software operation of an organization.

PO8: Apply standard Software Engineering practices and strategies in real-time software project development.

PO9: Design and develop computer programs/computer -based systems in the areas related to algorithms, networking, web design, cloud computing, IoT and data analytics.

PO10: Acquaint with the contemporary trends in industrial/research settings and thereby innovate novel solutions to existing problems

PO11: The ability to apply the knowledge and understanding noted above to the analysis of a given information handling problem.

PO12: The ability to work independently on a substantial software project and as an effective team member.

	Nam	e and Type of the Paper	Hrs	./w	eek	Total			
Subject/ Core Course Type Name		Nomo		т	P	Marks Por	UA	СА	Credits
oourse	туре	INDITIE	-	•	F	Paper			
BCA Sem-I									
	DSC1-1	Programming using C-I	2	-	-	50	30	20	2
	Practical	Practical based on DSC1-1	-	-	4	50	30	20	2
	DSC2-1	Python-I	2	-	-	50	30	20	2
	Practical	Practical based on DSC2-1	-	-	4	50	30	20	2
Major (Select any	DSC3-1	Operating System	2	-	-	50	30	20	2
three)	Practical	Practical based on DSC3-1	-	-	4	50	30	20	2
	DSC4-1	Office Automation	2	-	-	50	30	20	2
	Practical	Practical based on DSC4-1	-	-	4	50	30	20	2
	DSC5-1	Computational Mathematics	2	-	-	50	30	20	2
	Practical	Practical based on DSC5-1	-	-	4	50	30	20	2
SEC/ VSC	SEC-1	Basic Web Designing	2			50	30	20	2
	L1-1	English	2	-	-	50	30	20	2
AES, IKS, VEC	IKS	To be selected from the Basket of IKS		-	-	50	30	20	2
	VEC-1	Constitution of India	2	-	-	50	30	20	2
CC1	CC1	Community Engagement & Services	2	-	-	50	30	20	2
Total	1		16		12	550	330	220	22
		BCA Sem-II	.						
	DSC1-2	Programming using C-II	2	-	-	50	30	20	2
	Practical	Practical based on DSC1-2	-	-	4	50	30	20	2
	DSC2-2	Python-II	2	-	-	50	30	20	2
	Practical	Practical based on DSC2-2	-	-	4	50	30	20	2
Major (Select env)	DSC3-2	Software Engineering	2	-	-	50	30	20	2
(Select any three)	Practical	Practical based on DSC3-2	-	-	4	50	30	20	2
	DSC4-2	Linux and Shell Programming							
	Practical	Practical based on DSC4-2	-	-	4	50	30	20	2
	DSC5-2	Digital System							
	Practical	Practical based on DSC5-2	-	-	4	50	30	20	2
Generic/ Open Elective Courses	GE1/ OE1	e-Commerce	2	-	-	50	30	20	2
SEC/ VSC	SEC2	Advanced Web Designing	2			50	30	20	2
AES, IKS,	L1-2	English	2	-	-	50	30	20	2
VEC	VEC-2	Environmental science	2	-	-	50	30	20	2
CC2	CC2	Community Engagement & Services	2	-	-	50	30	20	2
	Т	otal	16	-	12	550	330	220	22
	Gran	nd Total	32	-	24	1100	660	440	44

Abbreviations:										
L: Lectures	T : Tutorials	P: Practical	UA : University Assessment	CA : College Assessment						
Generic/ Op	Generic/ Open Electives: GE/OE Skill Enhancement Courses: SEC									
Indian Know	ledge System:	IKS	Ability Enhancement Courses: AES							
Value Education Courses: VEC Vocational Skill and Skill Enhancement Courses: VSEC										
Co-curricula	r Courses: CC	;								

Student contact hours per week: 24 Hours (Min.)

Total Credits for BCA (Science)-I (Semester I and II): 44

Medium of instruction: English

- Practical Examination is the Semester wise after theory Examination. I.
- II. Duration of Practical Examination as per respective BOS guidelines.
- III. Separate passing is mandatory for Theory, Internal and Practical Examination.

Exit Option at Level 4.5 : Students can exit after Level 4.5 with under certificate course

inComputer

Programming if he/she complete the courses equivalent to a minimum of 44 credits and an additional. 4credits core NSQF course/Internship.

Course Structure:

Lectures and Practical's should be conducted as per the scheme of lectures and practical's indicated in thecourse structure.

Teaching and Practical Scheme

- I. Contact session for teaching 60 minutes each.
- II. One Practical Batch should be of 20 students.

Assessment

- The final practical examination will be conducted by the University appointed I. examiners internal as well as external at the end of the semester for each lab course and marks will be submitted to the university by the panel.
- II. The practical examination will be conducted semester-wise to maintain the relevance of the respective theory course with the laboratory course.
- III. The final examinations shall be conducted at the end of the semester.

Practical Examination:

- I. Each paper carries 30 Marks.
- II. **Duration of Practical Examination:** 2 Hrs.
- III. Nature of Question Paper: There will be four questions of 10 Marks each. Students will attempt any two out of four questions.
- IV. Certified Journal carries 5 Marks and Viva voce carries 5 Marks.

Standard of Passing:

- I. Minimum 12 marks in each subject. There shall be separate passing for theory (semesterend examand Internal) and practical also.
- II. Admission to BCA (Science) Part -II is allowed even if the student fails in all the subjects of part-I.
- III. Admission to BCA (Science) Part-III is allowed only if a student has passed on all thesubjects of BCA (Science) Part-I.

Board of Paper Setters /Examiners:

For each semester-end examination, there will be a board of Paper setters and examiners

for everycourse. While appointing paper setters/examiners, care should be taken to see that

there is at least one person specialized in each unit of the course.

Credit system implementation: As per the University norm's.

Fees Structure: As approved by the PAHS University fee fixation committee.

Intake Capacity: 60

Award of Class:

Grading: PAHS University has introduced a ten-point grading system as follows:

Sr. No.	Grade Abbreviation	From (Marks)	To (Marks)	Status	Grade Point	Description
1	0	80	100	Pass	10	Excellent / Outstanding
2	A+	70	79.99	Pass	9	Very Good
3	А	60	69.99	Pass	8	Good
4	B+	55	59.99	Pass	7	Fair
5	В	50	54.99	Pass	6	Above Average
6	C+	45	49.99	Pass	5	Average
7	С	40	44.99	Pass	4	Below Average
8	F	0	39.99	Fail	0	Fail

	BCA (Science)-I, Level - 4.5 UG Certificate Level							
	Sem: I							
Paper C	Paper Category: DSC1-1 (Major)							
Pape	r Name:	Programming	gusing	gC-I				
Credit: 02					Theor	י y:	2 Hrs./\	Veek
Marks:	UA	: 30		CA:	20		Total:	50

Course Objectives:

- 1. To understand the various steps in Program development.
- 2. To understand the basic concepts in C Programming Language.
- 3. To learn how to write modular and readable C Programs
- 4. To learn to write programs (using structured programming approach) in 'C' to solve problems
- 5. To build efficient programs in "C" language essential for future programming.

Course Outcomes:

- 1. Able to understand the basic concepts of C programming language.
- 2. Enhance skill on problem solving by constructing algorithms
- 3. Students will be able to comprehend the general structure of C program, concepts of variable, datatype, and operator and be able to create a C program to demonstrate these concepts.
- 4. Able to design and develop various programming problems using C programming concepts.
- 5. Understand and use various constructs of the programming language such as conditionals, iteration.
- 6. Demonstrate the use of strings and string handling functions
- 7. Apply skill of identifying appropriate programming constructs for problem solving.

Unit-I: Programming Methodology & Introduction to C: [15]

Programming Methodology: Definition of Problem , Problem solving steps , Programming planning tools, Definition of Logic, Types of logic- Sequence, Selection, Iteration, Algorithm- Definition, Characteristics, Examples of algorithm., Flowchart- Definition, Characteristics , flowchart symbols, Examples converts algorithms to flowchart, Pseudo Code- Definition, characteristics, Examples **Introduction to 'C':** History of 'C' language, Features of 'C' language, Application of C language, Structure of 'C' program, Compilation and execution of program. (Object File and Executable File), 'C' Fundamentals: 'C' tokens- Keywords , Identifier, Special symbols ('C' character sets), Variables, Constants, Data types- Primitive, Derived, User defined, Operators- Arithmetic, Logical, Assignment,

Relational, Bitwise, Conditional, Increment, Decrement, sizeof, comma operator, Conditional (ternary) operator, etc., Type casting or type conversion, typedef and enum., Data input and output operations: Introduction to input and output operations, Introduction to different header files and its use, stdio.h header file functions - printf(), scanf(), getchar(), putchar(), Different format specifier with their use, Different back slash (escape sequence) character constants with their use

Unit-II: Control Statements & Array:

[15]

Control Statements: Introduction to control statement, Types of control statements- Selective or Decision making –Two way decision making- if statement with its different forms, Multiway decision making-switch statement, Iterative or looping statement -While loop, do-while loop, for loop, Unconditional branching (jump) Statement - break, continue, goto.

Arrays: Introduction to array, Types of array -One dimensional, Two dimensional, Multidimensional array, Declaration and initialization of an array, Memory allocation of an array.

Character array (string): Declaration, Operation on string, Inbuilt String functions.

Reference Books

- 1. Programming in ANSI-C E. Balgurusamy, 8th edition, MCGraw Hill Publication
- 2. The C programming Language Ritchie and Kernighan, 2nd Edition, Pearson
- Programming through C Language, Dr. Tulashiram B. Pisal and Mr. Balasaheb J. Kshirsagar, InSc Publishing House(IPH)
- 4. Let Us C Y.C. Kanetkar, 15th edition, BPB Publication

	BCA (Science)-I, Level - 4.5 UG Certificate Level							
	Sem:							
Paper Ca	Paper Category: Practical (Major)							
Pape	r Name:	Practical bas	ed on D	DSC1	-1			
	Credit: 02 Practical: 4 Hrs./Week						Veek	
Marks:	UA	: 30	C	CA:	20		Total:	50

- 1. Write a program to check a given number is prime or not
- 2. Write a program to check a given number is palindrome or not
- 3. Write a program to check a given number is perfect or not
- 4. Write a program to check a given number is Armstrong or not
- 5. Write a program to check a given number is strong or not
- 6. Write a program to display prime numbers between 1 to 1000
- 7. Write a program to display the first 100 prime numbers.
- 8. Write a program to find the smallest and largest element from an array
- 9. Write a program to search for an element in an array
- 10. Write a program to display diagonal elements from a given matrix.
- 11. Write a program to display the transpose of a given matrix.
- 12. Write a program to display the addition and multiplication of two matrices.
- 13. Write a program to count vowels from a given string
- 14. Write a program to find the length of a given string without using the library function.

BCA (Science)-I, Level - 4.5 UG Certificate Level								
	Sem: I							
Paper Ca	Paper Category: DSC2-1 (Major)							
Pape	r Name:	Python-I						
Credit: 02 Theory:					י y :	2 Hrs./\	Veek	
Marks:	UA	A: 30		CA:	20		Total:	50

Course Objectives:

- 1. To learn the fundamentals of python Programming
- 2. To learn different data structures used in Python
- 3. To learn different control statements used in logic development.
- 4. To learn the various operations on the array, list, tuple, string, set, and dictionary.

Course Outcomes:

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

- 1. Understand the basic concepts and applications of Python.
- 2. Design, create, build, and debug python applications.
- 3. Explore Integrated Development Environment (IDE).
- 4. Write and apply decision structures for different operations.
- 5. Write loop structures to perform iterative tasks.

Unit-I: Introduction to Python & Control Statements in Python: [15]

Introduction: Features of Python, Python Virtual Machine, Memory management, Garbage Collection, Installation of Python, setting the path to operating system environment, writing the first Python program, executing a Python program., **Datatypes in Python:** Datatypes-Numeric, Sequence Type-String, List, Tuple, Boolean, Set, Dictionary, Binary Types, Type conversion- implicit and explicit, Python comments, literals, constants, Identifiers, naming conventions, operators, operator precedence and associativity, input and output statements, command-line arguments., **Control Statements:** if statement, if..else statement, if..elif..else statement, while loop, for loop, else suite, infinite loop, nested loops, word indentation, break statement, continue statement, pass statement, assert statement, return statement.

Unit-II: Sequence in Python, Array & Functions:

String, List, Tuple, Set and Dictionary: Creating string, manipulating different operations on string, creating list, manipulating different operations on list, list comprehensions, creating tuple, manipulating different operations on tuple, creating set, manipulating different operations on set, creating dictionary, manipulating different operations on dictionary.

Arrays in Python: Introduction, advantages of array, creating an array, types of arrays, importing array module, indexing and slicing on arrays, methods of array module.

Functions: Difference between function and method, defining a function, calling function, returning result from a function, returning multiple values from a function, functions are objects, formal and actual arguments, types of arguments, local, nonlocal and global variables, global keyword, recursive functions, anonymous functions or lambdas, using lambdas with filter(), map() and reduce() functions

Reference Books

- 1. Python: The Complete Reference by Martin C. Brown.
- 2. Core Python Programming, Dreamtech publications, by R. Nageswara Rao.
- 3. Python Programming, A modular approach, First Edition, Pearson, by Taneja Sheetal
- 4. Learning with Python, Dreamtech publications, by Allen Downey
- 5. Python Programming for the Absolute Beginner by Michael Dawson-Cengage Learning.

BCA (Science)-I, Level - 4.5 UG Certificate Level								
	Sem: I							
Paper Ca	Paper Category: Practical (Major)							
Pape	r Name:	Practical bas	ed on DSC2	2-1				
	Credit: 02 Practical: 4 Hrs./Week						Neek	
Marks:	UA	.: 30	CA:	20		Total:	50	

- 1. Write a Python program to find the sum of a list of numbers using a for loop.
- 2. Write a Python program to display stars in right angled triangular form using nested for loops.
- 3. Write a Python program to display a multiplication table from 1 to 10 using nested for loops.
- 4. Write a Python program to display numbers from 10 to 6 and break the loop when the number about to display 5.
- 5. Write a Python program to display numbers from 1 to 5 using the continue statement.
- 6. Write a Python program to find the first occurrence of substring in a given main string.
- 7. Write a Python program to display elements in a list in reverse order.
- 8. Write a Python program to accept elements in the form of a tuple and display their sum and average.
- 9. Write a Python program to create a dictionary with employee details and retrieve the values upon giving keys.

BCA (Science)-I, Level - 4.5 UG Certificate Level								
	Sem:							
Paper C	Paper Category: DSC3-1 (Major)							
Pape	r Name:	Operating Sy	stem					
Credit: 02					Theor	יץ:	2 Hrs./\	Veek
Marks:	UA	: 30		CA:	20		Total:	50

Course Objective:

Students will try to learn-

- 1. To understand the main components of an OS and their functions.
- 2. Describe the functions of a modern OS with respect to convenience, efficiency and the ability to evolve.
- 3. To make aware of different types of OS and their services.
- 4. To learn different process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.

Course Objectives:

- 1. To provide a sound understanding of the Computer operating system, its structures, and its functioning.
- 2. To understand the services provided by and the design of an operating system.
- 3. To understand different approaches to memory management.
- 4. To understand the services provided by and the design of an operating system.
- 5. To understand what a process is and how processes are synchronized and scheduled.

Unit I: Introduction to Operating system

Operating system: Definition of operating system, Types of Operating Systems-Batch, Multiprogramming, Time Sharing, Real-Time, Distributed, Parallel, OS Services, System components, System Calls., **Process Management:** Introduction to Process, Process states, Process Control Block, Context switching, Operations on Process, Introduction to Thread, Difference between process and thread, types of thread.

[10]

Unit II: Scheduling & Process Synchronization

Scheduling- Concept of Process Scheduling, Types of Schedulers, Scheduling criteria, Scheduling algorithms Preemptive and Non-preemptive, FCFS, SJF, Round Robin, Priority Scheduling, Multilevel Queue Scheduling, Multilevel- feedback Queue Scheduling.

Process Synchronization: The Producer Consumer Problem, Race Conditions, Critical Section Problem, Semaphores, Busy waiting, Spinlock, and Classical Problems of Synchronization: Reader-Writer Problem, Dinning Philosopher Problem.

Reference Books:

- 1. Operating System Concepts By Siberchatz and Galvin.
- 2. Modern O.S. By Andrews Tanenbaum.

BCA (Science)-I, Level - 4.5 UG Certificate Level								
	Sem:							
Paper Ca	Paper Category: Practical (Major)							
Pape	r Name:	Practical based	d on DSC3-1					
Credit: 02				Practica	al:	4 Hrs./\	Veek	
Marks:	UA	.: 30	CA	20		Total:	50	

- 1. Write a Python Program to implement the producer–consumer problem using semaphores.
- 2. Write a Python program to simulate the concept of the Dining-Philosophers problem.
- 3. Write a program to implement Threading and Synchronization Applications
- 4. Write a program for implementation of Priority scheduling algorithms
- 5. Write a program for implementation of Round Robin scheduling algorithms
- 6. Write a program for implementation of FCFS scheduling algorithms.
- 7. Write a program for implementation of SJF scheduling algorithms.
- 8. Write a program to simulate the concept of Dining-Philosophers problem.
- 9. Write a program to implement Threading and Synchronization Applications.
- 10. Write a program to implement banker's algorithm for deadlock avoidance.
- 11. Write a program to implement algorithm for deadlock detection.
- 12. Write a program for implementation memory allocation methods for fixed partition
- 13. Write a program to simulate the following contiguous memory allocation techniques
 - o Worst-fit b) Best-fit c) First-fit
- 14. Write a program to implement Paging technique for memory management.
- 15. Write a program for implementation of FIFO, LRU and LFU page replacement algorithm.

BCA (Science)-I, Level - 4.5 UG Certificate Level								
	Sem:							
Paper C	Paper Category: DSC4-1 (Major)							
Pape	r Name:	Office Autom	nation					
Credit: 02					Theor	y:	2 Hrs./\	Week
Marks:	UA	: 30		CA:	20		Total:	50

Course Objectives:

- 1. To provide an in-depth training in use of office automation, internet and internet tools. The course also helps the candidates to get acquainted with IT.
- 2. To help the students to understand how to format, edit, and print text documents and prepare for desktop publishing.
- 3. To create various documents newsletters, brochures, making document using photographs, charts, presentation, documents, drawings and other graphic images.
- 4. To work with the worksheet and presentation software..

Course Outcomes:

- 1. At the end of this course, the student should be able to
- 2. Integrate both graphs and tables created in Microsoft Excel into a laboratory report in Microsoft Word.
- 3. Generate equations, sample calculations, and basic diagrams in Microsoft Word.
- 4. Input experimental data into Microsoft Excel.
- 5. Perform calculations in Microsoft Excel using both manually inputting formulas and built-in Functions.
- 6. Generate simple and effective tables and graphs to describe experimental data in Microsoft Excel.
- 7. Properly format and organize a formal laboratory report in Microsoft Word.

Unit-I: Introduction to Computer & MS-Word :

Introduction to Computer: Applications of Computer, Advantages of Computer, Characteristics of Computer, Hardware and Software, Block diagram of computer

[15]

MS Word: Working with Documents -Opening and Saving files, Editing text documents, Inserting, Deleting, Cut, Copy, Paste, Undo, Redo, Find, Search, Replace, Formatting page and setting Margins,

Converting files to different formats, Importing and Exporting documents, Sending files to others, Using Tool bars, Ruler, Using Icons, using help., **Formatting Documents:** Setting Font styles, Font selectionstyle, size, color etc, Type face - Bold, Italic, Underline, Case settings, Highlighting, Special symbols, Setting Paragraph style, Alignments, Indents, Line Space, Margins, Bullets and Numbering.. **Setting Page style:** Formatting Page, Page tab, Margins, Layout settings, Paper tray, Border and Shading, Columns, Header and footer, Setting Footnotes and end notes – Shortcut Keys; Inserting manual page break, Column break and line break, Creating sections and frames, Anchoring and Wrapping, Setting Document styles, Table of Contents, Index, Page Numbering, date and Time, Author etc., Creating Master Documents, Web page. Creating Tables: Table settings, Borders, Alignments, Insertion, deletion, Merging, Splitting, Sorting, and formula. , Drawing: Inserting Clip Arts, Pictures/Files etc., **Tools:** Word Completion, Spell Checks, Mail merge, Templates, Creating contents for books, Creating Letter/Faxes, Creating Web pages, Using Wizards, Tracking Changes, Security, Digital Signature. Printing Documents – Shortcut keys.

Unit-II: MS-Excel & MS Power point:

[15]

MS Excel: Spread Sheet and its Applications, Opening Spreadsheet, Menus - main menu, Formula, Editing, Formatting, Toolbars, Using Icons, Using help, Shortcuts, Spreadsheet types. Working with Spreadsheets- opening, Saving files, setting Margins, Converting files to different formats (importing, exporting, sending files to others), Spread sheet addressing - Rows, Columns and Cells, Referring Cells and Selecting Cells – Shortcut Keys., Entering and Deleting Data: Entering data, Cut, Copy, Paste, Undo, Redo, Filling Continuous rows, columns, Highlighting values, Find, Search and replace, Inserting Data, Insert Cells, Column, rows and sheets, Symbols, Data from external files, Frames, Clipart, Pictures, Files etc, Inserting Functions, Manual breaks., Setting Formula: Finding total in a column or row, Mathematical operations (Addition, Subtraction, Multiplication, Division, Exponentiation), using other Formulae., Formatting Spreadsheets: Labeling columns and rows, Formatting- Cell, row, column and Sheet, Category- Alignment, Font, Border and Shading, Hiding/ Locking Cells, Anchoring objects, Formatting layout for Graphics, Clipart etc., Worksheet Row and Column Headers, Sheet Name, Row height and Column width, Visibility - Row, Column, Sheet, Security, Sheet Formatting and style, Sheet background, Colour etc, Borders and Shading ,Shortcut keys., Working with sheets: Sorting, Filtering, Validation, Consolidation, and Subtotal., Creating Charts: Drawing. Printing. Using Tools - Error checking, Formula Auditing, Creating and Using Templates, Pivot Tables, Tracking Changes, Security, Customization.

MS Power point: Presentation – Opening new presentation, Different presentation templates, setting backgrounds, selecting presentation layouts., **Creating a presentation:** Setting Presentation style,

Adding text to the Presentation. Formatting a Presentation: Adding style, Colour, gradient fills, Arranging objects, Adding Header and Footer, Slide Background, Slide layout. Adding Graphics to the Presentation- Inserting pictures, movies, tables etc into presentation, Drawing Pictures using draw., **Adding Effects to the Presentation:** Setting Animation and transition effect. Printing Handouts, Generating Standalone Presentation viewer.

Reference Books:

- 1. Information Technology in Business: Principles, Practices, and Opportunities by James A Senn, Prentice Hall.
- 2. Technology and Procedures for Administrative Professionals by Patsy Fulton-Calkins, Thomson Learning.
- 3. Computer Fundamental MS Office-Including Internet and Web Technology: Anupama Jain, Avneet Mehra
- 4. The Complete Reference: Virginia Andersen, McGraw Hill
- 5. MS Office 2007 in a Nutshell: S. Saxena, Vikas Publications
- 6. MS-Office 2007 Training Guide: S. Jain, BPB Publications
- 7. Learning Computer Fundamentals, MS Office and Internet and Web Technology: D. Mai dasani. Reading, Vols. 1 and 2. Macmillan, 1975, Bhasker, W. W. S. and Prabhu, N. S.

BCA (Science)-I, Level - 4.5 UG Certificate Level								
	Sem:							
Paper C	Paper Category: Practical (Major)							
Раре	r Name:	Practical based	d on DS	SC4-1				
Credit: 02 Practi					Practica	al:	4 Hrs./\	Veek
Marks:	UA	: 30		CA:	20		Total:	50

1. Create a news-paper document with at least 200 words,

a. Use margins as, top:1.5, bottom:2, left:2, right:1 inches.

- b. Use heading "Gandhi Jayanti", font size: 16, font color: red, font face: Arial Black.
- c. With first letter "dropped" (use drop cap option) of the first paragraph containing a picture at the right side
- d. Use three columns from the second paragraph onwards till the half of the page.
- e. Then use heading "Computer basics"
- f. Create paragraph using two columns till the end of the page.
- 2. Create a Mathematical question paper using, at least five equations
- a. With fractions, exponents, summation function
- b. With at least one "m*n" matrix
- c. Basic mathematical and geometric operators.
- d. Use proper text formatting, page color and page border.
- 3. Create a flowchart using,
- a. Proper shapes like ellipse, arrows, rectangle, and parallelogram.
- b. Use grouping to group all the parts of the flowchart into one single object.
- 4. Create a table using table menu with,
- a. At least 5 columns and 10 rows.
- b. Merge the first row into one cell.
- c. Merge the second row into one cell, then split the second row into three cells.
- d. Use proper table border and color.
- e. Insert proper content into the table with proper text formatting.
- 5. Create a table using two columns,
- a. The left column contains all the short-cut keys and right side column contains the function of the short-cut keys.
- b. Insert a left column using layout option. Name the heading as Serial No.
- 6. Create two letters with the following conditions in Ms Word and find the difference.
- a. Write a personal letter to your friend using at least 100 words and two paragraphs. The date must be in top- right corner. Use "justify" text alignment and 1.5 line spacing for the body of the letter. Letter must containproper salutation and closing.
- b. Use step by step mail-merge wizard to design a letter. (Mailing step by step mail merge wizard letters start from a template select template letters select proper template create new document OK)
- 7. Create a letter, which must be sent to multiple recipients.
- a. Use Mail-Merge to create the recipient list.
- b. Use excel sheet to enter the recipient.
- c. Start the mail merge using letter and directory format. State the difference.

List of Assignments (MS Excel)

- 1. Create a table "Student result" with following conditions.
- a. The heading must contain, Sl. No., Name, Mark1, Mark2, Mark3, Total, average and result with manual entry.
- b. Use formulas for total and average.
- c. Find the name of the students who has secured the highest and lowest marks.
- d. Round the average to the nearest highest integer and lowest integer (use ceiling and floor functionrespectively).
- 2. Do as directed
- a. Create a notepad file as per the following fields Slno. name th1 th2 th3 th4 th5 total % grade
- b. Import this notepad file into excel sheet using "data from text" option.
- c. Grade is calculated as, i. If %>=90, then grade A ii. iii. If %>=80 and <90, then grade B If %>=70 and <80, then grade C iv. If %>=60 and <70, then grade D v. If %<60, then grade F

Item	Year1	Year2	Year3	Year4
Item1	1000	1050	1100	1200
Item2	950	1050	1150	1200
Item3	1100	1200	1200	1300

- 3. Create a sales table using the following data,
- a. Draw the bar-graph to compare the sales of the three items for four years using insert option.
- b. Draw a line-graph to compare the sales of three items for four years using insert option.
- c. Draw different pie-charts for the given data using insert option.
- d. Use condition, to highlight all the cells having value >=1000 with red color (use conditional formatting).

List of Assignments (MS PowerPoint)

- 1. Create a power-point presentation with minimum 5 slides.
- a. The first slide must contain the topic of the presentation and name of the presentation.
- b. Must contain at least one table.
- c. Must contain at least 5 bullets, 5 numbers.
- d. The heading must be, font size:32, font-face: Arial Rounded MT Bold, font-color: blue.
- e. The body must be, font size: 24, font-face: Comic Sans MS, font-color: green. f. Last slide must contain "thankyou".
- 2. Create a power-point presentation with minimum 10 slides 23
- a. Use word art to write the heading for each slides.
- b. Insert at least one clip-art, one picture
- c. Insert at least one audio and one video
- d. Hide at least two slides
- 3. Create a power-point presentation with minimum
- a. 5 slides
- b. Use custom animation option to animate the text; the text must move left to right one line at a time.

List of Assignments (MS Access)

- 1. Create a database "Student" with,
- a. At least one table named "mark sheet" with field name "student name, roll

number, mark1, mark2, mark3, mark4, total"

- b. The data types are, student name: text, roll number: number, mark1 to mark4: number, total: number. Rollnumber must be the primary key.
- c. Enter data in the table. The total must be calculated using update query.
- d. Use query for sorting the table according to the descending/ascending order of the total marks.
- 2. With addition to the table above,
- a. Add an additional field "result" to the "mark sheet" table.
- b. Enter data for at least 10 students c. Calculate the result for all the students using update queries, if total>=200,then pass, else fail.
- c. Search the students, whose name starts with "sh".
- d. Show the names and total marks of the students who have passed the examination.

BCA (Science)-I, Level - 4.5 UG Certificate Level									
	Sem:	I							
Paper Category: DSC5-1 (Major)									
Pape	r Name:	Computational Mathematics							
	Credit:	02	Theory: 2 Hrs.			2 Hrs./\	Veek		
Marks:	UA	. 30		CA:	20		Total:	50	

Course Objectives -

The aim of this course is to prepare learners for a foundational understanding of computers, encompassing:

- 1. To introduce the concepts of mathematical logic.
- 2. To introduce the concepts of relations, counting and functions.
- 3. To perform the operations associated with counting, functions, and relations.
- 4. To use Graph Theory for solving problems.
- 5. To introduce the matrix and its operations

Course Outcomes-

- 1. Ability to apply mathematical logic to solve problems.
- 2. Understand sets, relations, counting, matrix, and graph.
- 3. Able to use logical notation to define and reason about fundamental mathematical concepts such as relations, counting, matrices and graphs.
- 4. Able to model and solve real-world problems using graphs and trees.

Unit I: - Logic and Proofs & Relations

[15]

Logic and Proofs: Propositional logic, Applications of Propositional logic, propositional equivalences, Predicates and Quantifiers, Rules of inference.

Relations: Relations and their properties, Representing relation, Closures of relations, Partial orderings.

Unit II: - Counting, Graphs and Matrices:

Counting: The basics of counting, the pigeonhole principle, Permutation and Combinations, Applications of recurrence relations, solving recurrence relations, Divide and Conquer algorithms and recurrence relations.

Graphs: Graphs and Graphs models, Graph terminology and special types of graphs, Representing graphs and Graph isomorphism, Connectivity, shortest path problems.

Matrices: Introduction, operations, inverse, Rank of a matrix, solution of simultaneous linear equations, Eigen values and Eigen Vectors.

Reference Books:

- 1. Modern Algebra S. Arumugam and A. Thangapandi Isaac, Scitech publications, 2005.
- 2. Invitation to Graph Theory- S.Arumugam and S.Ramachandran, Scitech Publications, 2005, Chennai.
- 3. Discrete Mathematical Structures with applications to Computer Science Tremblay and Manohar, McGrawHill, 1997.
- 4. Mathematical Structure for Computer Science, Discrete Mathematics and its Applications, Judith L.Gersting, W.H. Freeman and Company, Seventh Edition, 2014.

BCA (Science)-I, Level - 4.5 UG Certificate Level									
	Sem:	Ι							
Paper Ca	ategory:	P	Practical (Major)						
Pape	r Name:	Pr	Practical based on DSC5-1						
	Credit:	02			Practical:		al:	4 Hrs./Week	
Marks:	UA	٨:	30	CA: 20 Total: 50					50

- 1. Write Python program for AND, OR, NOT, IMPLICATION, and BICONDITIONAL operation to Implement basic propositional logic operations.
- 2. Write a program to check the equivalence of two propositional expressions using truth tables.
- 3. Write a Python program to check if a relation is reflexive, symmetric, transitive, and antisymmetric and to verify if a relation is a partial order and find the Hasse diagram of a partial order.
- 4. Write a program to represent a relation as a matrix and perform basic operations on the matrix.
- 5. Write Python program to count permutations, combinations, and apply the principle of inclusionexclusion and also generate permutations and combinations of a given set.
- 6. Write a program to solve problems using the pigeonhole principle (e.g., finding duplicates in a list).
- 7. Write a Python program to solve simple recurrence relations (e.g., Fibonacci sequence) and analyze their complexity.
- 8. Write Python functions to solve linear homogeneous recurrence relations with constant coefficients.
- 9. Write Python programs for merge sort and quicksort, and analyze their time complexity using recurrence relations.
- 10. Write a Python program to represent graphs using adjacency lists and adjacency matrices.
- 11. Write a Python program to check if two graphs are isomorphic using adjacency matrices.
- 12. Write Python programs for breadth-first search (BFS), depth-first search (DFS), and Dijkstra's algorithm to find the shortest path.
- 13. Write Python functions to perform matrix addition, subtraction, multiplication, finding the transpose and inverse and rank of a matrix.
- 14. Write a Python program to solve systems of linear equations using Gaussian elimination and LU decomposition.
- 15. Write a Python program to compute the eigenvalues and eigenvectors of a matrix using NumPy.

BCA (Science)-I, Level - 4.5 UG Certificate Level									
	Sem:	I							
Paper Category: SEC1									
Pape	r Name:	Basic Web Designing							
	Credit:	02	Theory:			2 Hrs./Week			
Marks:	UA	: 30	CA: 20 Total: 50					50	

Course Objective:

- 1. Give the distinguishing characteristics of scripting language.
- 2. Discuss the reasons for and effects of nonstandard client-side scripting language characteristics, such as limited data types, dynamic variable types and properties, and extensive use of automatic type conversion.
- 3. Develop event-driven programs that use HTML intrinsic event attributes, DOM events, listeners, and DOM-generated events.
- 4. Use the DOM to modify a document's attributes and style properties as well as to modify its parse-tree representation.

Course Outcomes:

- 1. Explain the history of the internet and related internet concepts that are vital in understanding web development.
- 2. Discuss the insights of internet programming and implement complete applications over the web.
- 3. Demonstrate the important HTML tags for designing static pages and separate design from content using Cascading Style sheet.
- 4. Utilize the concepts of JavaScript.

Unit I: Basic Web Design

Introduction to Web Design: Introduction to Networking, Introduction to Internet, Applications of the Internet. Introduction to HTML, Structure of HTML, Creating and opening HTML file, Singular and paired tags, Text formatting tag, Anchor tag, Lists, Image, Image Map, Table, Frames and Frameset, form.

[10]

Unit II: CSS and JavaScript

Introduction to CSS and JavaScript: Introduction to CSS, Types of CSS, Use of CSS, Selectors, Properties, Values.

CSS Properties: Background, Text, Fonts, Link, List, Table, Box Model, Border, Margin, Padding, Display, Positioning, Floating, Opacity, Media type, Backgrounds, Animations, Multiple Column Layout, Navigation bar.

Introduction to JavaScript- JavaScript Variables and Data types, Operators, Built-in functions in JavaScript, Control structure in JavaScript, JavaScript Objects- Object, Array, String, Date, Math, Number, Boolean Introduction to Java Script DOM and BOM, difference between DOM and BOM, DOM Objects- Document Object, Element Objects, Node Objects, BOM Objects-Window, Navigator, Location, Document, user-defined function, Form Validation, event and event handling in JavaScript.

Reference Books:-

- 1. HTML5 Black Book Kogent Learning Solutions Inc Dream-tech.
- 2. Beginning JavaScript and CSS Development with jQuery Richard York.
- 3. Beginning HTML and CSS Rob Larsen

BCA (Science)-I, Level - 4.5 UG Certificate Level									
	Sem:	II							
Paper Category: DSC1-2 (Major)									
Pape	r Name:	Programming using C-II							
	Credit:	02		Theor	y:	2 Hrs./\	Veek		
Marks:	UA	30		CA:	20		Total:	50	

Course Objective:

- 1. To develop logic which will help them to create programs, applications in 'C' language.
- 2. To learning the basic programming constructs they can easily switch over to any other language in future.
- 3. To learn problem solving techniques using C.
- 4. To train the student in the basic concepts of the programming language C.
- 5. To improve the programming skills using C.

Course Outcomes:

Upon successful completion of this course, students will be able to-

- 1. To implement advance C programming concepts like function, pointer, structure and union etc.
- 2. To understand the dynamics of memory by the use of pointers.
- 3. To understand the file handling using C Programming language.
- 4. To understand the concept of macros and preprocessor.

Unit-I: Functions & Pointers:

Functions: Introduction to function, Need or use of function, Types of Functions, Inbuilt/ Predefined/ Library functions, User defined function, Steps to add or include user defined function in program-Function declaration (Prototyping) ,Function calling , Function definition (Function Implementation), **Types of Function:** Function with argument without return value, Function with argument with return value, Function without argument with return value, Function without argument without return value, local and global variable, Recursion and its Advantages and disadvantages.

Pointers: Definition and declaration of pointer, Pointer initialization, Pointer and function, Pointer and array, Chain of pointer, Call by value, Call by reference, Dynamic memory allocation, malloc(), calloc(), realloc() and free(), **Storage Classes:** Introduction to storage Classes and use, storage classes- auto, extern, static, register

[20]

Unit-II: Structures and Union & File Handling:

Structures and Union: Definition and declaration of structure and union, difference between structure and union, Array of structures, Passing structure to function, Pointer to structure, Nested structure, selfreferential structure., **File Handling:** Introduction, Standard input- get char(), getch(), getche(), Standard output- putchar(), putch(), putche(), Formatted input- scanf(), sscanf(), fsclose(), File modes, Text and binary mode., **Macros and Preprocessing:** Features of C pre-processor, Macro – Declaration, Expansion, File Inclusion

Reference Books

- 1. Programming in ANSI C, E. Balagurusamy, McGraw Hill Education India Private Limited.
- 2. Programming through C Language, Dr. Tulashiram B. Pisal and Mr. Balasaheb J. Kshirsagar, InScPublishing House (IPH), Edition-1st, 2021.
- 3. The 'C' programming language, Brian Kernighan, Dennis Ritchie, Pearson.

[10]

BCA (Science)-I, Level - 4.5 UG Certificate Level								
	Sem:	II						
Paper Category: Practical (Major)								
Pape	r Name:	Practical based on DSC1-2						
	Credit:	02		Practical:			Neek	
Marks:	UA	.: 30	CA:	CA: 20 Total: 50				

- 1. WAP to find out the factorial of any number.
- 2. Write a program that swaps two numbers using pointers.
- 3. Write a program in which a function is passed the address of two variables and then alters its contents.
- 4. Write a program that 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 the area and circumference from the main() function.
- 5. Write a program that counts the total number of vowels present in the string.
- 6. Write a program to find the sum of n elements entered by the user.
- 7. Write a program to allocate memory dynamically using malloc() and calloc().
- 8. Write a program to illustrate the difference between structure and union.
- 9. WAP to pass an array of structure to function.
- 10. Write a program to copy the content of one file into another file.
- 11. Write a program to display the content of binary files.
- 12. Write a program to accept integer numbers in file, find even and odd numbers between them store even numbers into even file and odd number into odd file and display the content of files.

BCA (Science)-I, Level - 4.5 UG Certificate Level									
	Sem:	II							
Paper Category: DSC2-2 (Major)									
Pape	r Name:	Pythor	n-II						
	Credit:	02			Theory:			2 Hrs./Week	
Marks:	UA	: 30			CA:	20		Total:	50

Course Objective:

- 1. To learn the use of functions in programming.
- 2. To understand the use of modules and packages in the application hierarchy.
- 3. To understand python programming using the object-oriented programming principles.
- 4. To learn handling of various exceptions during the application development.
- 5. To understand the working with different file operations.

Course Outcomes:

- 1. Upon successful completion of this course, students will be able to-
- 2. Write and implement a functional and modular approach to application development.
- 3. Design an application using object-oriented paradigm.
- 4. Create error free applications by applying the exception handling concept.
- 5. Design an application that contains the use of different files for data processing.

Unit-I: Object Oriented programming:

Modules and packages: Introduction to modules and packages, import statement, from...import statement, creating our own modules, working with built-in modules. Math module, time module and random module.

Python Object Oriented: Difference between procedure-oriented and object-oriented programming. Features of object-oriented programming- classes and objects, inheritance, polymorphism, encapsulation, abstraction. Creating class, self-variable, constructor, types of variables, namespaces, types of methods, passing member of one class to another class, inner classes. Types of inheritance, super() method, method overloading, method overriding, abstract classes, and interfaces.

Unit-II: Threading, Exception Handling & File:

Threading: Understanding threads, Class and threads, Creating Threads, Thread Synchronization, Treads Life cycle, Multi-threading.

[15]

[15]

Exception Handling: Error in Python program, exceptions, steps in exception handling using try, except, else and finally blocks, types of exceptions- built-in and user-defined exceptions, assert statement. File Input Output: Types of files in Python, opening a file- the file opening modes, closing a file, working with text files containing strings, working with binary files, with statement, pickling and unpickling, seek() and tell() methods, random accessing of binary files, zipping and unzipping files, working with directories.

Reference Books

- 1. Python: The Complete Reference by Martin C. Brown.
- 2. Core Python Programming, Dreamtech publications, by R. Nageswara Rao.
- 3. Python Programming, A modular approach, First Edition, Pearson, by Taneja Sheetal

BCA (Science)-I, Level - 4.5 UG Certificate Level									
	Sem:	=							
Paper Category: Practical (Major)									
Pape	r Name:	Pi	Practical based on DSC2-2						
	Credit:	02			Practical:		al:	4 Hrs./\	Veek
Marks:	UA	۹:	30	CA: 20 Total: 50					50

- 1. Write a function to return the addition and subtraction of two numbers using a function return two values.
- 2. Write a python program to demonstrate the different methods of array module.
- 3. Write a python program to demonstrate the types of array.
- 4. Write a python program to understand the positional arguments of a function
- 5. Write a python program to understand the keyword arguments of a function
- 6. Write a python program to understand the default arguments in a function
- 7. Write a python program to understand Variable length arguments in a function.
- 8. Write a python program to understand Anonymous (lambda) Function.
- 9. Write a python program to understand local, non-local and global variables.
- 10. Write a python program to create a module and import it.
- 11. Write a python program to create a package and import it.
- 12. Write a python program to demonstrate the instance method, class method and static method.
- 13. Write a python program to demonstrate inner classes.
- 14. Write a python program to demonstrate Constructors in Inheritance.
- 15. Write a python program to demonstrate method overloading.
- 16. Write a python program to demonstrate method overriding.
- 17. Write a python program to read all the strings from the text file and display them.
- 18. Write a python program to append data to an existing file and display them.
- 19. Write a python program to count a number of lines, words and characters in a file.
- 20. Write a python program to copy an image file into another file.
- 21. Write a python program to apply different manipulation operations of directories.
- 22. Write a python program to handle the ZeroDivisionError exception.
- 23. Write a python program to handle syntax errors given by eval() function.
- 24. Write a python program to handle IOError produced by open() function.
- 25. Write a python program to illustrate the use of raising an exc

BCA (Science)-I, Level - 4.5 UG Certificate Level								
	Sem:	II						
Paper Category: DSC3-2 (Major)								
Pape	r Name:	Software Engineering						
	Credit:	02	Theory: 2 Hrs./V			Veek		
Marks:	UA	: 30		CA:	20		Total:	50

Course Objectives –

The aim of this course is to prepare learners for a foundational understanding of computers, encompassing:

- 1. To understand the fundamental concepts and characteristics of systems.
- 2. To categorize different types of systems and understand system analysis.
- 3. To explore various SDLC models and their applications with different functional and non-functional requirements.
- 4. To learn various techniques for gathering system requirements.
- 5. To understand the process of designing and implementing a system.
- 6. To understand coding standards, size measures, complexity analysis, and verification.
- 7. To grasp the fundamentals of software testing.
- 8. To learn about software implementation and the maintenance process.

Course Outcomes-

- 1. Students will be able to define a system, identify its characteristics, and describe various types of systems.
- 2. Students will be able to describe and compare models such as the Waterfall model, V-shape model, Spiral model, Prototyping, Incremental, RAD, and Agile methodologies.
- 3. Students will be able to identify, document, and analyze user and system requirements.
- 4. Students will be proficient in conducting interviews, questionnaires, record reviews, and observations for requirement gathering.
- 5. Students will be able to design data flow diagrams, entity-relationship diagrams, structured charts, and create a data dictionary. They will also learn input and output design.
- 6. Students will learn to write clean, maintainable code, estimate effort and cost, and verify code correctness.

- 7. Students will be able to perform different types of testing and understand testing methodologies.
- 8. Students will understand the steps involved in implementing software and the different types of software maintenance.

[15]

Unit I: - Introduction to Software Engineering

System concepts: Introduction system, characteristics, Elements of system, Types of system, System Analysis, Role of System Analyst, Software Engineering: Definition, Characteristics of software, Qualities of software. System Development life cycle: Waterfall model, V-shape model, Spiral model, Prototyping, incremental, RAD, Agile. Software requirements: Functional, Non-functional requirements, User requirement, System requirements, Fact finding techniques: Interviews, Questionnaire, Record reviews, Observation Analysis and Design Tools: Flow charting, Decision tables, Decision Trees, Structured English, Structure charting Techniques.

Unit II: - System Design and Implementation, Maintenance [15]

Data flow Diagram (Physical, Logical), Entity relation diagram, structured chart, Data Dictionary, Input and output design, Types of Dependencies, Normalization (1NF,2NF,3NF,BCNF,4NF,5NF) Coding: Verification, size measures, complexity analysis, coding standards, Effort Estimation, Cost Estimation, Testing fundamentals Construction of the system: traditional and incremental approaches, conversion methods, Software Implementation, Overview of maintenance process, types of maintenance.

Reference Books-

- 1. Analysis and Design of Information Systems By James Senn.
- 2. Practical guide to structure System Design By Miller/Page/jones.
- 3. Software Engineering By Pressman. 4. System Analysis and Design By Parthsarty

BCA (Science)-I, Level - 4.5 UG Certificate Level									
	Sem:	II							
Paper Category: Practical (Major)									
Pape	r Name:	Practical bas	Practical based on DSC3-2						
	Credit:	02		Practical:		al:	4 Hrs./\	Veek	
Marks:	UA	A: 30	CA: 20 Total: 50					50	

Use Appropriate Tools for Solving the practical's

- 1. Perform a system analysis for a given case study. Identify and document the system's components, processes, and interactions.
- 2. Create a checklist or template to evaluate the qualities of software (correctness, reliability, efficiency, usability, maintainability, portability) for a given application.
- 3. Create flowcharts / diagrams for different SDLC models (Waterfall, V-shape, Spiral, Prototyping, Incremental, RAD, Agile).
- 4. Develop a requirements specification document for a small software project, including functional, non-functional, user, and system requirements.
- 5. Document requirements for an online shopping system.
- 6. Design a questionnaire and conduct interviews or observations to gather requirements for a proposed system. Summarize your findings in a report.
- 7. Create a flowchart for a simple process using any tool.
- 8. Develop a decision table and decision tree for a given set of business rules.
- 9. Develop a structure chart for a software module using a tool. Explain the hierarchy and interactions between modules.
- 10. Create a DFD and ERD for a Library Management System using a tool. Include entities, attributes, and relationships and create a data dictionary for a small database system. Include definitions, data types, and relationships for all data elements. Also design input forms and output reports for a given system.
- 11. Normalize a set of database tables to 1NF, 2NF, 3NF, BCNF, 4NF, and 5NF.
- 12. Use techniques like COCOMO or Function Point Analysis to estimate the effort and cost for a small software project. Document your estimation process and results.
- 13. Develop test cases for a given software module. Perform unit testing, integration testing, and system testing using tools like JUnit or Selenium.
- 14. Design a conversion plan for migrating data from an old system to a new system. Discuss parallel, direct, phased, and pilot conversion methods.
- 15. Develop a deployment plan for a software application. Include steps for installation, configuration, and user training and write an overview of the software maintenance process.

Describe different types of maintenance (corrective, adaptive, perfective, preventive) and provide examples for each type.

BCA(Science)-I, Level - 4.5 UG Certificate Level								
	Sem:	II						
Paper C	ategory:	DSC4-2 (Ma	DSC4-2 (Major)					
Pape	r Name:	Linux and Shell Programming						
	Credit:	02		Theory:			2 Hrs./Week	
Marks:	UA	.: 30	CA: 20 Total: 50					

Course Objective: -

Students will try to learn:

- 1. To introduce Basic Linux general purpose Commands
- 2. To learn different editor
- 3. To learn shell script concepts.
- 4. To learn file management and permission advance commands.
- 5. To learn awk, grap, perl scripts.

Course Outcomes: -

Students will be able to:

- 1. Identify the basic Linux general purpose commands.
- 2. Apply and change the ownership and file permissions using advance Linux commands.
- 3. Use the awk, grep, perl scripts.
- 4. Implement shell scripts.
- 5. Apply basic of administrative task.

Unit 1: - Introduction of Linux

History of Linux, Architecture of Linux system & features, Kernel, Shell & its type, Difference between Windows and Linux. Linux Distributions, Working environments: KDE, GNOME, Xface4, Hardware requirement, Installation procedure of Linux, Create partitions, Configuration of X system Users & Groups Management:- Create Users, Create groups, Special groups, Assigning permissions to users and Groups, File and Directory permissions- chmod, chown, chgrp., Linux File System:-Hierarchy of File system, File System parts- Boot Block, Super Block, Inode, Block, Data Block, File types, Devices and Drives in Linux, Mounting devices (CD/DVD, usb, hard drive partition), file system

(10)

Unit II: - Linux Command and Shell Programming

I/O and Redirection, Piping, Linux commands File and directory Management Commands:-mkdir, rmdir, cd and pwd, file, ls, cat, more, less, File and Directory Operations: find, cp, mv, rm, ln etc, Printing the files - lpr, lpq, lprm etc., Filter Commands & Editor:- Filters: head, tail, pr, cut, paste, sort, uniq, tr, grep, egrep, fgrep, sed., Communication commands:- mesg, talk, write, wall, mail., Text Editors- vi, vim, Archive and File compression commands, Process Management: Shell process, Parent and children, Process status, System process, Multiple jobs in background and foreground, Changing process priority with nice. Listing processes, ps, kill, premature termination of process.

Introduction to shell scripting, Writing and executing simple shell scripts, Variables, data types, and operators in shell scripting, Meta characters, Control structure, Loop structure and case statement, Writing and using shell functions, passing arguments to shell scripts and functions, Returning values from functions,

Reference Books :

- 1. Official Red Hat Linux Users guide by Redhat, Wiley Dreamtech India
- 2. Beginning Linux Programming by Neil Mathew & Richard Stones, Wiley Dreamtech India
- 3. Red Hat Linux Bible by Cristopher Negus, Wiley Dreamtech India
- 4. UNIX Shell Programming by Yeswant Kanethkar, BPB
- 5. Shell Scripting: Expert Recipes for Linux, Bash, and More" by Steve Parker
- 6. Classic Shell Scripting" by Arnold Robbins and Nelson H.F. Beebe
- 7. Learning the bash Shell: Unix Shell Programming" by Cameron Newham and Bill Rosenblatt

BCA (Science)-I, Level - 4.5 UG Certificate Level									
	Sem:	II							
Paper Category: Practical (Major)									
Pape	r Name:	Practical based on DSC4-2							
	Credit:	02		Practical: 4 H			4 Hrs./\	s./Week	
Marks:	UA	30	30 CA: 20 Total: 50						

- 1. Navigate the file system using commands such as ls, cd, pwd, mkdir, and rmdir.
- 2. View file contents using commands such as cat, less, and head.
- 3. Manipulate files and directories using commands such as cp, mv, rm, and touch.
- 4. Use commands such as chmod and chown to change file permissions and ownership.
- 5. View running processes and their resource usage using commands such as ps, top
- 6. Manage processes using commands such as kill, killall, pkill, and pgrep.
- 7. Manage users and groups using commands such as useradd, usermod, groupadd, and passwd.
- 8. Write and execute a simple shell script using a text editor and the bash interpreter.
- 9. Use variables to store and manipulate data within a shell script.
- 10. Implement conditional statements in shell.
- 11. Pass arguments to functions and return values using the return statement.
- 12. Read input from files and process data using commands such as cat, grep, awk, and sed.
- Use redirection operators (>, >>, <) to redirect input and output streams between files and commands.
- 14. Write Linux script for checking given number is prime, Armstrong and palindrome.
- 15. Write Linux script to display Fibonacci sequence up to n numbers.

BCA (Science)-I, Level - 4.5 UG Certificate Level									
	Sem:	-							
Paper Ca	ategory:	D	SC5-2 (Ma j	jor)					
Pape	r Name:	D	igital Syster	m					
	Credit:	0	2			Th	eory:	2 Hrs./\	Neek
Marks:	UA	\ :	30		CA:	20		Total:	50

Course Objectives:

- 1. To provide a comprehensive introduction to the fundamental axioms, theories and conventions underlying the operation of digital systems.
- 2. To equip students with the necessary skills which will allow them to analyse, design, test, and simulate the operation of basic digital circuits.
- 3. To utilize a variety of digital logic design and simulation tools.

Course Outcomes:

At the end of this course, the student should be able to

- 1. Identify a digital system and its main characteristics, and differentiate between digital and analogue systems
- 2. Describe the concepts of binary numbers and binary encoding, and perform conversions between binary, decimal, and hex numbers and between binary codes
- 3. Perform basic mathematical operations using binary numbers, and design digital systems capable of performing such operations.
- 4. Describe theorems and axioms of Boolean Algebra, and utilize them effectively in the process of designing digital systems.
- 5. Model, analyse, design, test, and simulate the operation of combinational and sequential circuits using analytic and modular methodologies and tools.
- 6. Explain the concept of memory in digital systems, and design basic memory modules.

[15]

Unit I: Basic of Digital System

History and Overview: History, applications of digital systems, digital signals and analog **igns** advantages and disadvantages of digital systems.

Number System: The binary, decimal and hexadecimal number system, conversion of binary, decimal and hexadecimal number system, Signed and unsigned binary numbers, representing signed binary

numbers using the 2's complement method, performing basic mathematical operations using binary numbers (addition, subtraction, multiplication, division), design of adder and subtractor circuits. **Block diagram –** ALU, memory unit, control unit, motherboard, SMPS, expansion slots, serial and parallel ports.

Unit II: Boolean Algebra & Microprocessor

[15]

Boolean Algebra fundamentals: definition of boolean algebra and logic gates, types of logic gates (AND, OR, NOT, NAND, NOR, XOR, XNOR), DeMorgan's Theorem, implementation of logic circuits using boolean equation and truth table.

Microprocessor: History and overview, comparative study of 8085, 8086 and Pentium processor, architecture of Pentium microprocessor, features of Pentium microprocessor, applications of Pentium, addressing modes, instruction set, types of programming languages, assembly language programming, applications of Pentium.

Reference Books:

- 1. Digital Systems: Principles and Applications by Ronald J. Tocci, Neal S. Widmer, and Greg Moss
- Digital Design: With an Introduction to the Verilog HDL, VHDL, and SystemVerilogby M. Morris R. Mano, and Michael D. Ciletti
- 3. Fundamentals of Digital Logic with VHDL Design, by Stephen Brown and Zvonko Vranesic
- 4. Digital Fundamentals, by Thomas L. Floyd
- 5. Digital Logic Design, by B. Holdsworth
- 6. System Programming by John J. Donowon
- 7. System programming and Operating System by Dhamdhere
- 8. System Software by Beck

BCA (Science)-I, Level - 4.5 UG Certificate Level								
	Sem:	1						
Paper (Category:	Practical (Major)						
Paper Name:		Practical based on DSC5-2						
Credit:		02 Practical: 4 Hrs./Week				eek		
Marks:	UA	A: 30		CA:	20		Total:	50

Tools / Software: NASM (Netwide Assembler) / MASM (Microsoft MacroAssembler) / TASM (Turbo Assembler) etc.

- 1. Write an ALP to add 2 Multibyte no.
- 2. Write an ALP to subtract two Multibyte numbers.
- 3. Write an ALP to multiply two 16-bit numbers.
- 4. Write an ALP to divide two numbers.
- 5. Write an ALP to multiply two ASCII no.
- 6. Develop and execute and assembly language program to perform the conversion fromBCD to binary.
- 7. Write an ALP to convert binary to BCD.
- 8. Write an ALP to find the square of a number.
- 9. Write an ALP to find the cube of a number.
- 10. Write an ALP to separate odd and even numbers
- 11. Write an ALP to separate positive and negative numbers
- 12. Write an ALP to find logical ones and zeros in a given data
- 13. Write an ALP to demonstrate the AND Gate, OR Gate,
- 14. Write an ALP to demonstrate the NOT Gate, NAND Gate
- 15. Write an ALP to demonstrate the NOR Gate, XOR Gate.

BCA (Science)-I, Level - 4.5 UG Certificate Level										
Sem:										
Paper Category:			SEC2							
Paper Name:		Advance Web Designing								
Credit:		02			Theory: 2			2 Hrs./V	2 Hrs./Week	
Marks:	UA	٨:	30		CA:	20		Total:	50	

Course Objective:

- 1. To identify the capabilities of JavaScript and jQuery and their role in web design and thedocument object model.
- 2. To respond to user events using jQuery, creating interactivity.
- 3. To provide a collection of syntax for template designs.
- 4. To develop and apply appropriate website or web application information architectures.
- 5. To design effective user interfaces.

Course Outcomes:

Upon successful completion of this course, students will be able to-

- 1. Understand the concepts of jQuery, bootstrap.
- 2. Build interactive web applications using JQuery and bootstrap.
- 3. Develop solution to complex problems using appropriate method, technologies, frameworks, web services and content management
- 4. Extend this knowledge to .Net Platforms, Java Technologies, Full Stack Development.

Unit I: jQuery

jQuery: Introduction to jQuery, Difference between document Ready and Window Load, Adding jQuery to Your Web Pages, jQuery Syntax, jQuery Selectors, jQuery Event Methods, Effects-Hide and Show, Fading, Sliding, Animation, Button in jQuery, How to handle forms, Callback Functions, Chaining, Get and Set Content and Attributes, Add Elements, Add Several New Elements, Remove Elements, Get and Set CSS Classes, css() Method, The noConflict() Method

[12]

Unit II: Bootstrap

Bootstrap: Introduction to Bootstrap Framework, History of Bootstrap, Advantages, Responsive web page, Major Features of Bootstrap, Setting up Environment, Syntax of Bootstrap, Introduction to Bootstrap container and its types, Introduction to Bootstrap Grid, Bootstrap Forms: Form Layouts, Inline Form, form controls (input, textarea, checkbox, radio, select), Static Control, Media Objects, Filters., **Bootstrap Components:** Jumbotron, Button, Grid, Table, Alert, Wells, Badge and label, Panels, Input Groups, Input Types, Modals, Popover, Scrollspy, Pagination, Pager, Image, Glyphicon, Carousel, Progress Bar, List Group, Dropdown, Collapse, Tabs/Pill, Navbar,

Reference Books:

- Nicholas C Zakas, "Professional JavaScript for Web Developers", 3rd Edition, Wrox/WileyIndia, 2012.
- David Sawyer Mcfarland, "JavaScript & jQuery: The Missing Manual", 1st Edition, O'Reilly/Shroff Publishers & Distributors Pvt Ltd, 2014.
- 3. Benjamin Jakobus and Jason Marah, Mastering Bootstrap 4, Packt Publishing, 2016.
- Jacob Lett, Bootstrap 4 Quick Start: Responsive Web Design and Development Basics forBeginners, 2018.

BCA (Science)-I, Level - 4.5 UG Certificate Level									
	Sem:								
Paper Category:			GE1/OE1						
Paper Name:		e-Commerce							
Credit:		02			Theory: 2 Hrs./Week			Veek	
Marks:	UA	۹:	30		CA:	20		Total:	50

Course Objectives:

- 1. Increasing significance of E-Commerce and its applications in Business and Various Sectors.
- 2. Mechanism of business transactions through electronic media.
- 3. To identify Latest Trends in E-Commerce and Digital Marketing, along with Opportunities.
- 4. Payment transactions in a secured network.
- 5. Different modes of E-Commerce like Electronic data interchange.

Course Outcomes:

After Completion of the subject student should able to

- 1. Analyze the impact of E-commerce on business models and strategy
- 2. Describe the major types of E-commerce.
- 3. Explain the process that should be followed in building an E-commerce presence.
- 4. Identify the key ethical, social, and security threats in the E-commerce environment.
- 5. Understand the implementation of Electronic Data Interchange (EDI) in day to day life

Unit I: Introduction to E-commerce

Introduction to e-commerce, Functions of ecommerce, comparison of traditional commerce and ecommerce, advantages and disadvantages of e-commerce Introduction to e-commerce business models – B2B,B2C,C2C and scope of e- commerce, Benefits of e-commerce, applications of e-commerce, **Electronic Data Interchange(EDI)**-Concept of EDI, Difference between paper based business and EDI based business. Application area of EDI, Action plan for implementing EDI, And factors influencing the choice of EDI software, **Electronic Commerce Environment and Opportunities:** Background, The Electronic Commerce Environment, Electronic Marketplace Technologies. Modes of Electronic Commerce:

[15]

Electronic Data Interchange, Migration to Open EDI, Electronic Commerce with www/Internet, Commerce Net Advocacy, web Commerce Going Forward.

Unit II: E-Marketing & E-Finance:

[15]

E-Marketing: Introduction to traditional marketing, comparison of traditional marketing and E-marketing, Impact of e-commerce on market, Marketing issues in E-marketing Promoting your e-business, direct marketing, one to one marketing, **E-Finance:** Meaning of finance, areas of E –Finance, operations in E-Banking - Trading Operations in stock Market trading Importance, advantages of E-trading Master Card/Visa Secure Electronic Transaction: Introduction, Business Requirements, Concepts, payment Processing. E-Mail and Secure Email Technologies for Electronic Commerce: Introduction, The Means of Distribution, A model for Message Handling, E-mail working, Multipurpose Internet Mail Extensions, Message Object Security Services, Comparisons of Security Methods, MIME and Related Facilities for EDI over the Internet., **E-Commerce in India:** Status of E-commerce in India , Problems and opportunities in E-commerce in India, Future of E-commerce and legal issues involved in e- commerce in india

Reference Books

- 1. E-commerce by C.S.V. Murthy
- 2. E-commerce by Bajaj and Nag
- 3. WebCommerce Technology Handbook, by Daniel Minoli, Emma Minoli, McGraw-Hill
- 4. E-Commerce fundamentals and applications Hendry Chan, Raymond Lee, Tharam Dillon.
- 5. E-commerce by Lauden

Equivalent Subject for Old Syllabus B.C.A. (Science) - I (Semester-I and II)

Semester-I							
Sr.	Name of the Old Paper	Name of the New Paper					
No.	(w.e.f. 2022-2023)	(w.e.f. 2024-2025)					
1.	Fundamental of Computer	No Equivalence					
2.	Office Automation	Office Automation					
3.	Programming and Problem Solving using 'C' – I	Programming using 'C'-I					
4.	Web Programming-I	Basics Web Designing					
5.	Basics of Mathematics	No Equivalence					
6.	Descriptive Statistics	No Equivalence					
7.	Fundamentals of Electronics	No Equivalence					
8.	Linear Electronics	No Equivalence					
Semester-II							
1.	Introduction to Python Programming	Python-I (Sem-I)					
2.	Operating System	Operating System (Sem-I)					
3.	Programming and Problem Solving using 'C' – II	Programming using 'C'-II					
4.	Web Programming-II	Advanced Web Designing					
5.	Graph Theory	No Equivalence					
6.	Probability Theory	No Equivalence					
7.	Digital Electronics	No Equivalence					
8.	Introduction of Microprocessor and Interfacing	No Equivalence					

(NEP-2020)