

BIDAR UNIVERSITY BIDAR



BACHELOR OF COMPUTER APPLICATION SYLLABUS FOR UNDER GRADUATE STATE EDUCATION POLICY-2024

**WITH EFFECT FROM
THE ACADEMIC YEAR 2026-27**

Submitted by

Sri. Srikanth Doddamani

Associate Professor in Computer Science, KASCC, Bidar

Chairman

Board of Studies (UG)

Bidar University Bidar

Curriculum Structure for B.C.A w. e. f 2025-26

Category	Course Code	Title of the Paper	Marks			Hour/ Week	Credits	Duration of Exams (Hrs.)
			IA	Semester End Exam (SEE)	Total			
SEMESTER V								
DSC05	THEORY							
	Skill-2T	Research Methodology	10	40	50	2	2	2
	SEPBCAT13	Python Programming	20	80	100	4	4	3
	SEPBCAT14	Statistical Computing and R Programming	20	80	100	4	4	3
	SEPBCAT15	Software Engineering	20	80	100	4	4	3
	PRACTICALS							
	SEPBCAP13	Python Programming Lab	10	40	50	4	2	3
	SEPBCAP14	R Programming Lab	10	40	50	4	2	3
	SEPBCAP15	Software Engineering Lab	10	40	50	4	2	3
TOTAL : HOURS/CREDIT					500	26	20	
SEMESTER VI								
DSC06	THEORY							
	SEPBCAT16	PHP\My SQL	20	80	100	4	4	3
	SEPBCAT17	Digital Image Processing	20	80	100	4	4	3
	SEPBCAT18	Data Science	20	80	100	4	4	3
	PRACTICALS							
	SEPBCAP16	PHP\My SQL Lab	10	40	50	4	2	3
	SEPBCAP17	Digital Image Processing Lab	10	40	50	4	2	3
	Skill-3 MP	Major Project(90 for Project Evaluation, 30 for Viva-Voce=120,30 for IA	30	120	150	6	3	3
TOTAL : HOURS/CREDIT					550	28	19	

Year	III	Course Code : SkillT-2		Credits	2
Semester	V	Course Title : Research Methodology		Hours	30
Course Pre-requisites, if any		NA			
Formative Assessment Marks : 10		Summative Assessment Marks : 50	Duration of SEE: 2 hrs.		
Course Objectives		After completing this course satisfactorily, a student will be able to: - To provide the students the basic tools and materials for research - To acquaint them with the process of research - To guide them in presenting their research and results - To supervise the students as they write research papers on topics related to their Ph.D. research			
Unit No.	Course Content			Hours	
UNIT I	Meaning of research; objectives of research; basic steps of research; criteria of good ,research; types of research, Meaning of research problem; selection of research problem. Review of related literature- Meaning, necessity and sources. Hypothesis- Meaning, function and types of hypothesis; Null/Alternative hypothesis Variables- Meaning and types.			15	
UNIT II	Research design: Types of research design- exploratory, descriptive, diagnostic and experimental. Sampling- Meaning and types of sampling; Probability and Non-Probability. Tools and techniques of data collection- questionnaire, schedule, interview, observation, case study, survey etc. Statistics and its significance in research. Research reports: Writing preliminaries, main body of research, references and bibliography; Meaning and importance of workshop, seminar, conference, symposium etc. in research. Plagiarism- Concept and significance of plagiarism			15	
Recommended Learning Resources					
Text Books:					
1. C.R. Kothari, “Research Methodology Methods and Techniques (2 nd Edition)” , Vishwa Prakashan,2006					
2. Donald H. McBurney, “Research Methods(5 th edition)”, Thomson Learning, ISBN: 81-315-0047-0, 2006					
Reference Books:					
1. Donald R.Cooper, Pamela S. Schhindler, Business Research Methods, 8th Edition, Tata MCGRAW-Hill Co.Ltd.,2006.					
2. Fuzzy Logic with Engg Applications, Timothy J.Ross, Wiley Publications, 2nd Edition. R. Panner Selvam, Research Methodology					

E-Resources:

Top Research Methodology e-Resources

1. SAGE Publications – Ranjit Kumar's Companion Sites

- **Fifth Edition Resource Hub:** Offers a step-by-step guide through eight stages of research, from problem formulation to report writing.
 - Includes videos, journal articles, case studies, and interactive checkpoints.
 - Two learning pathways: one for clarifying concepts and another for deeper exploration.
 - Access it [here](#).
- **Fourth Edition Companion Site:** Tailored for beginners with no prior research experience.
 - Features free tools for students and instructors, including examples and operational steps.
 - Access it [here](#).

2. NCRM (National Centre for Research Methods) Online Tutorials

- Over 100 free tutorials covering:
 - **Statistical analysis and modelling**
 - **Qualitative and creative methods**
 - **Ethics and research design**
- Created by expert methodologists, these include videos, datasets, worksheets, and interactive graphics.
- Ideal for both independent learners and educators.
- Explore tutorials [here](#).

Year	III	Course Code : SEPBCAT13		Credits	4
Semester	V	Course Title : Python Programming		Hours	60
Course Pre-requisites, if any		NA			
Formative Assessment Marks : 20		Summative Assessment Marks : 80	Duration of SEE: 3 hrs.		
Course Objectives		After completing this course satisfactorily, a student will be able to: - Understand the fundamentals of Python syntax, semantics, and structure. - Learn core programming concepts: variables, data types, operators, control flow, and functions. - Explore Python’s built-in data structures: lists, tuples, dictionaries, and sets. - Gain familiarity with object-oriented programming (OOP) in Python.			
Unit No.	Course Content			Hours	
UNIT I	Introduction: Features and Applications of Python, Python Versions, Installation of Python, Python Command line mode and IDEs, Simple Python program, Identifiers, Keywords, Statements and Expressions, Variables, Indentation, Comments.			10	
UNIT II	program structure, output function, input function ,data types, type operators and expressions, Control Flow: Types of Control Flow Statements-if, if-else, elif, while loop, break, continue Statements ,for loop, range() and exit() functions.			10	
UNIT III	Data structures – str, list, tuple, dict, set, user defined functions - variable number of arguments, default parameters, key value pairs as arguments, Data Storage, Data formatting.			15	
UNIT IV	Classes, Objects, Inheritance-Types of Inheritance, Errors and Exception Handling, Modules, Packages, Regular Expressions, File Handling.			15	
UNIT V	GUI Programming-tkinter Module-Windows and widgets, Layout Management-pack, grid, place, Relational databases-SQLite. Introduction to Data Visualization, Matplot Library, Different types of charts using Pyplot: Line Chart, BarChart , Histogram and Pie Chart.			10	
Recommended Learning Resources					
Reference Books:					
1. Bill Lubanovic,” Introducing Python- Modern Computing in Simple Packages”, O’Reilly Publication					
2. Allen Downey, Jeffrey Elkner, Chris Meyers, “How to Think Like a Scientist –Learning with Python(1 st Edition”, Green Tea Press, 2002.					
3. Charles Dierbach, “Introduction to Computer Science Using Python”, Wiley Publication.					
4. Allen B. Downey, “Think Python-How to think like a computer scientist(1 ST Edition)”, O’Reilly Publication.					

E-Resources:

Interactive Learning

1. **Python.org Official Tutorial** – The official and comprehensive beginner tutorial.
2. **W3Schools Python Tutorial** – Interactive examples and quizzes.
3. **LearnPython.org** – Hands-on browser-based coding exercises.
4. **Codecademy: Learn Python** – Guided projects and interactive lessons.

Video Tutorials

1. **freeCodeCamp.org Python Course (YouTube)** – 4-hour full beginner's course.
2. **Programming with Mosh - Python for Beginners** – Very clear and structured.

Free Books

1. **Automate the Boring Stuff with Python** – Practical projects for beginners.
2. **Think Python** – Free eBook that teaches programming concepts through Python.

Practice & Challenges

1. **HackerRank – Python Track**
2. **LeetCode Python Problems**
3. **Exercism – Python Track** – Practice with mentoring support.

Tutorials & Guides

1. **Real Python** – In-depth tutorials and code walkthroughs.
2. **GeeksforGeeks Python Programming** – Great for algorithms and data structures.
3. **Python Tutor** – Visualize code execution step-by-step.

Advanced & Specialization

Machine Learning / Data Science

1. **Kaggle Learn Python** – Short hands-on data science lessons.
2. **DataCamp** – Paid, but has excellent data-focused Python courses.
3. **fast.ai** – Deep learning with Python (free).

Web Development & Automation

1. **Flask Mega-Tutorial by Miguel Grinberg** – Learn web development with Flask.
2. **Django Documentation** – For building full web apps.
3. **Selenium with Python Docs** – Browser automation.

Year	III	Course Code : SEPBCAT14		Credits	4
Semester	V	Course Title : Statistical Computing and R Programming		Hours	60
Course Pre-requisites, if any		NA			
Formative Assessment Marks : 20		Summative Assessment Marks : 80		Duration of SEE : 3 hrs.	
Course Objectives	After completing this course satisfactorily, a student will be able to: • Learn to use R and RStudio for statistical computing. • Understand R syntax, data types, and control structures. • Use R data structures like vectors, matrices, lists, and data frames. • Visualize data using R’s plotting systems (base, ggplot2). • Import/export data from various formats (CSV, Excel, databases). • Perform and interpret different distribution using R • Implement regression models, hypothesis testing, and ANOVA.				
Unit No.	Course Content			Hours	
UNIT I	Introduction: History of R , Features of R, Applications of R, Advantages and Disadvantages of R, R Basic Syntax, Comments, Keywords, Data Types, Data Structures-Vector, List, Array, Matrices, Data Frame, Operators.			15	
UNIT II	.Special values, Classes, Coercion, basic plotting, Statements and Loop-If, If Else, Else If ,Switch, Next and Break Statement, For loop, While Loop, Repeat Loop, Reading Files, Writing Files, Exception, Functions.			10	
UNIT III	Statistics and Probability: Basics, Mean, Median, Mode, Standard Deviation, Variance, Common Probability Distribution: Common Probability Mass Function-Bernoulli, Binomial, Poisson, Common Probability Density Functions-Uniform, Normal Student’s T-Distribution.			10	
UNIT IV	Statistical Testing and Modeling, Sampling Distribution, Hypothesis Testing, Components of hypothesis testing, Proportion Testing, Error in Hypothesis Testing, Power of Hypothesis Testing, Analysis of Variance(ANOVA)			10	
UNIT V	Simple Linear Regression, Multiple Linear Regression, Advanced Graphics-Plot Customization, Plotting Regions and Diagnostics, Point and Coordinate interaction, Customizing Traditional R Plots, Specialized Text and Label Notation, Click Defining Colors and Plotting in Higher Dimensions, Representing and using Color, 3D Scatter Plots.			15	
Recommended Learning Resources					
Text Books:					
1. Maria Dolores Ugarte , Ana F. Militino , Alan T. Arnholt “Probability and Statistics with R (2 nd Edition), CRC Press, 2016.					
2. P. Dalgaard.” Introductory Statistics with R(2 nd Edition). (Springer 2008)					
Reference Books:					
1. Michael Akritas, " Probability & Statistics with R for Engineers and Scientists”, 2nd Edition on, CRC Press, 2016.					
2. Purohit, S.G.; Gore, S.D. and Deshmukh, S.R. (2015). Statistics using R, second edition. Narosa Publishing House, New Delhi.					

E-Resources:

Foundational Resources

- **The R Project for Statistical Computing** This is the official site for R. It includes downloads, manuals, FAQs, and links to CRAN (Comprehensive R Archive Network), which hosts thousands of R packages.
- **Free Resources for Learning R – Statistical Horizons** Offers curated free resources for beginners and intermediate learners. Includes video tutorials, guides, and links to seminars tailored for users transitioning from other statistical software like SPSS or Stata.

Practice-Based Learning

- **R Programming Exercises – w3resource** A hands-on platform with exercises ranging from basic syntax to advanced statistical techniques. Each exercise includes a sample solution, making it ideal for self-paced learning.

Courses and Tutorials

- **Coursera & edX** Search for courses like “R Programming” by Johns Hopkins University or “Data Science: R Basics” by Harvard. These platforms offer structured learning paths with certificates.
- **DataCamp & Codecademy** Interactive coding environments that teach R through bite-sized lessons and real-world projects.

Books and Manuals

- **“R for Data Science” by Hadley Wickham & Garrett Grolemund** Available online for free, this book is a go-to for learning tidyverse and data analysis workflows.
- **“The Art of R Programming” by Norman Matloff** A deeper dive into R as a programming language, ideal for those with some coding background.

Community and Support

- **Stack Overflow & RStudio Community** Great for troubleshooting and learning from real-world questions.
- **R-bloggers** Aggregates blog posts from R users around the world, covering tutorials, package updates, and case studies.

Year	III	Course Code : SEPBCAT15		Credits	4
Semester	V	Course Title : Software Engineering		Hours	60
Course Pre-requisites, if any		NA			
Formative Assessment Marks : 20		Summative Assessment Marks : 80	Duration of SEE : 3 hrs.		
Course Objectives	After completing this course satisfactorily, a student will be able to: - To comprehend the various software process models. - To understand the types of software requirements and SRS document. - To know the different software design and architectural styles. - To learn the software testing approaches .				
Unit No.	Course Content			Hours	
UNIT-I	Introduction to Software Engineering: The evolving role of software, Changing Nature of Software, Software myths ,Software Process Process models: The waterfall model, Incremental process models, Evolutionary process models, The Unified process, Agility and Agile Process model,			15	
UNIT-II	Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document. Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.			10	
UNIT-III	System models: Context Models, Behavioral models, Data models, Object models, structured methods. UML Diagrams. Software Project Planning: Size Estimation, Cost Estimation, COCOMO Model, COCOMO II Model.			10	
UNIT-IV	Design Engineering: Design process and Design quality, Design concepts, the design model. Modularity-Module Coupling, Module Cohesion, Strategy of Design : Bottom-Up Design, Top-Down Design, Hybrid Design, Function Oriented Design, Object Oriented Design			10	
UNIT-V	Testing Strategies: A strategic approach to software testing, Some Terminologies –Error, Mistake, Bug, Fault, and Failure, Levels of Software Testing, Unit Testing, Integration Testing, System Testing, Acceptance Testing, Functional Testing- Boundary Value Analysis, Equivalence Class Testing, Decision Table Based Testing, Cause effect Graph.			15	
Recommended Learning Resources					
Text Books:					
1. Roger S Pressman, ”Software engineering A practitioner’s Approach(6 th edition) McGraw Hill International Edition.					
2. Ian Sommerville, “Software Engineering(7 th edition), Pearson education.					
Reference Books:					
1. Pankaj Jalote, “Software Engineering, A Precise Approach”, Wiley India, 2010.					
2. Waman S Jawadekar, “Software Engineering: A Primer”, Tata McGraw-Hill, 2008					
3. Rajib Mall, “Fundamentals of Software Engineering”, PHI, 200					

E-Resources:

GeeksforGeeks – Software Engineering Tutorial

- Offers a structured guide covering software development models, architecture, testing, and maintenance.
- Great for beginners and intermediate learners.
- Includes examples, diagrams, and explanations of key concepts like requirements analysis, software evolution, and design principles.
- Visit: [GeeksforGeeks Software Engineering Tutorial](#)

The Forage – Best Online Resources for Software Engineering Students

- Curated list of top tools and platforms for learning software development.
- Includes resources for coding practice, interview prep, and real-world simulations.
- Ideal for students and professionals looking to upskill or switch careers.
- Visit: [The Forage Software Engineering Resources](#)

TutorialsPoint – Useful Software Engineering Resources

- Offers concise tutorials and links to additional reading material.
- Covers topics like software lifecycle, project management, and quality assurance.
- Good for quick reference and brushing up on specific topics.
- Visit: [TutorialsPoint Software Engineering Resources](#)

Bonus Tools & Platforms to Explore

- **GitHub** – Explore open-source projects and collaborate with other developers.
- **Stack Overflow** – Ask questions and learn from community-driven answers.
- **Coursera & edX** – Take structured courses from top universities like MIT and Stanford.
- **LeetCode & HackerRank** – Practice coding problems and prepare for technical interviews.

Year	III	Course Code : SEPBCAP13	Credits	2
Semester	V	Course Title : Python Programming Lab	Hours	50
Course Pre-requisites, if any				
Formative Assessment Marks :10		Summative Assessment Marks : 40	Duration of SEE : 3 hrs.	

PART-A :

1. Write a Python program to perform Arithmetic operation.
2. Write a Python program to find largest of three numbers.
3. Write a Python program to find the square root and cube root of a number.
4. Write simple Python program to demonstrate use of 'if ... else' statement
5. Write a python program to find factorial of a given number using functions
6. Write Python program to perform following operations on Lists:
 - a) Create list
 - b) Access list
 - c) Update list (Add item, Remove item)
 - d) Delete list
7. Create a list and perform the following methods
 - a) insert()
 - b) remove()
 - c) append()
 - d) len()
 - e) pop()
 - f) clear()
8. Create a dictionary and apply the following methods
 - a) Print the dictionary items
 - b) access items
 - c) use get()
 - d) change values
 - e) use len()

PART-B:

1. Create a tuple and perform the following methods
 - a) Add items
 - b) len()
 - c) check for item in tuple
 - d) Access items
2. Write a Python program to perform the following string operation.
 - a) lower()
 - b) upper()
 - c) find()
 - d) replace()
 - e) count()
 - f) len()
3. Write a Python program to perform the operations on sets.
4. Write a Python program to display month, and year by using built in python modules.
5. Write a Python program to print student information using class.
6. Demonstrate a python code to print try, except and finally block statements
7. Write a program in Python to demonstrate following operations:
 - a. Simple inheritance
 - b. Multiple inheritance.
8. Write python program to create GUI using tkinter Module.

Instructions:

1. A certified journal is mandatory for every student to appear for the practical examination.
2. The student has to execute a minimum of eight programs from each part to complete the lab course.

Year	III	Course Code : SEPBCAP14	Credits	2
Semester	V	Course Title : R Programming Lab	Hours	50
Course Pre-requisites, if any				
Formative Assessment Marks : 10		Summative Assessment Marks : 40	Duration of SEE : 3 hrs.	

PART-A:

1. Write a program for different types of data structure in R.
2. Write a R program that includes variables and data types.
3. Write a R program to demonstrating the use of arithmetic operators.
4. Write a R program R to demonstrating if-else, for loop, and while loop.
5. Write a R program to demonstrate default values for arguments.
6. Write a R program for calculating cumulative sums, and products, minima, maxima.
7. Write a R program that include linear algebra operation on vector and matrices.
8. Write R program for any visual representation of as an object with creating graphs using graphics functions: Plot(), Hist(), Linechart(), pie(), BoxPlot, ScatterPlots().

PART-B:

1. Write a program to create an any application of Linear Regression in a multivariate context for predictive purpose
2. Write a R program to calculate the factorial of a number.
3. Write a R program to create a sequence of numbers from 20 to 50 and find the mean of numbers from 20 to 60 and sum of numbers from 51 to 91.
4. Write a R program to create a vector which contains 10 random integer values between -50 and +50.
5. Write a R program to get the first 10 Fibonacci numbers.
6. Write a R program to print the numbers from 1 to 100 and print "Fizz" for multiples of 3 , print "Buzz" for multiples of 5 and print "Fizz Buzz" for multiples of both.
7. Write a R program to get the unique elements of a given string and unique numbers of vector.
8. Write a R program to create three vectors a,b,c with 3 integers. Combine the three vectors to become a 3X3 matrix where each column represents a vector. Print the content of the matrix.

Instructions:

1. A certified journal is mandatory for every student to appear for the practical examination.
2. The student has to execute a minimum of eight programs from each part to complete the lab course.

Year	III	Course Code : SEPBCAP15	Credits	2
Semester	V	Course Title : Software Engineering Lab	Hours	50
Course Pre-requisites, if any				
Formative Assessment Marks : 10		Summative Assessment Marks : 40	Duration of SEE : 3 hrs.	
1. State the activities of Software Engineering. 2. Illustrate SDLC Model. 3. Illustrate the Waterfall Model. 4. Illustrate Increment Process Model. 5. Illustrate Spiral Model. 6. Explain Requirement Engineering Process. 7. What is UML? Write categories of UML diagram. 8. What is ER-Diagram? Write notations of ER Diagram. 9. What is DFD? Write notation of DFD. 10. Describe Error, Fault, and Failure. 11. What is Software Testing? Explain different types of Software Testing.				

Instructions:

1. A certified journal is mandatory for every student to appear for the practical examination.

Year	III	Course Code : SEPBCAT16		Credits	4
Semester	V	Course Title : PHP MySQL		Hours	60
Course Pre-requisites, if any		NA			
Formative Assessment Marks : 20		Summative Assessment Marks : 80		Duration of SEE : 3 hrs.	
Course Objectives	After completing this course satisfactorily, a student will be able to: • Design dynamic and interactive web pages and websites. • Run PHP scripts on other server and retrieve results. • Handle databases like MySQL using PHP in websites.				
Unit No.	Course Content			Hours	
UNIT-I	Introduction to PHP: Introduction to PHP, History and features of PHP, Installation and Configuration of PHP, Embedding PHP code in your web pages, Understanding PHP,HTML ,Writing Comments in PHP, Sending Data to the Web Browser, Data Types in PHP, Keywords in PHP, Using Variables, Constants in PHP, Expressions in PHP, Operators in PHP.			10	
UNIT-II	Programming with PHP: Conditional statements: if, if-else, switch, The? Operator, Looping statements: while, do-while, for loop. Arrays in PHP: Introduction-What is Array?, Creating Arrays, Accessing array elements, Types of Arrays: Indexed v/s Associative arrays, Multidimensional arrays, Using Array Functions, Including and Requiring Files-use of Include() and Require, Implicit and Explicit Casting in PHP.			15	
UNIT-III	Using Functions: Functions in PHP, Function definition, Creating and invoking user-defined functions, Formal parameters versus actual parameters, Function and variable scope, Recursion, Library functions Date and Time Functions. String in PHP: What is String? , Creating and Declaring String, String Functions.			10	
UNIT-IV	Class-Objects, Forms in PHP: What is Class & Object, Creating and accessing a Class and Object, Object Properties, Object Methods, Overloading, Inheritance, Constructor and Destructor. Form Handling: Creating HTML form, Handling HTML Form data in PHP Database Handling Using PHP with MySQL: Introduction to MySQL: Database terms, Data Types.			15	
UNIT-V	Accessing MySQL- Using MySQL Client and Using PHP MyAdmin, MySQL Commands, Using PHP with MySQL: PHP MySQL Functions, Connecting to MySQL and Selecting the Database, Executing Simple Queries, Retrieving Query Results, Counting Returned Records, Updating Records with PHP			10	
Recommended Learning Resources					
Reference Books:					
1. Larry Ullman, “PHP & MySQL for Dynamic Web Sites(4 th Edition)”. 2. Robin Nixon, “Learning PHP, MySQL and JavaScript”, O’REILLY Publications. 3. Rasmus Lerdorf, Kevin Tatroe, Peter MacIntyre, “Programming PHP “,O’REILLY Publications.					

E-Resources:

Free / Budget-Friendly Resources

- Great Learning – “*Free Course: PHP Coding with MySQL*” – Covers PHP syntax, data types, arrays, strings, and then how to integrate PHP with MySQL. mygreatlearning.com
- Noble Desktop – Has a page listing free intro PHP & MySQL content, good step-in point. nobledesktop.com
- Eduonix Learning Solutions – “Learn PHP & MySQL Development From Scratch” (approx 4.5 hrs) – gives a quick boot-up into PHP + MySQL. eduonix.com
- Free YouTube/tutorials – e.g., full tutorial “PHP & MySQL Tutorial – Learn PHP From Scratch” on YouTube.

Paid / Comprehensive Courses

- Udemy – “PHP & MySQL for Beginners” course. Great for building from zero. [Udemy](https://www.udemy.com/)
- Udemy – “PHP for Beginners: The Complete PHP MySQL PDO Course” – covers PDO (modern database access in PHP). [Udemy](https://www.udemy.com/)
- Edureka – “PHP & MySQL with MVC Frameworks Certification Training” – goes beyond basics into frameworks / MVC architecture. edureka.co

Recommendation for You

Since you’re based in Karnataka, India, you probably want resources that:

- Start at a beginner level (if you’re new)
- Include local context / are affordable
- Offer practical project-based learning

So here’s a learning path:

1. Start with a free course (Great Learning or Eduonix) so you get the basics of PHP syntax and MySQL.
2. Move to one of the paid comprehensive courses when you’re comfortable, especially one with projects (e.g., Udemy or Edureka) to build real-world apps.
3. As you progress, make sure you practice by building small projects: e.g., a registration/login system, CRUD interface (Create/Read/Update/Delete) with PHP & MySQL.
4. Also learn to connect PHP to MySQL via **PDO** (rather than older `mysql_*` functions) since that’s modern/secure practice. The PDO-focused course above helps with that.
5. Try to deploy one of your PHP/MySQL apps on a live host (or on your local with something like XAMPP) to understand how it works in real conditions.

Year	III	Course Code : SEPBCAT17			Credits	4
Semester	VI	Course Title : Digital Image Processing			Hours	60
Course Pre-requisites, if any		NA				
Formative Assessment Marks : 20		Summative Assessment Marks : 80		Duration of SEE: 3 hrs.		
Course Objectives		After completing this course satisfactorily, a student will be able to: - Understand image formation and the role human visual system plays in perception of grey and color image data. - Describe various applications of image processing in various sectors like medical, defense, etc. - Learn the signal processing algorithms and techniques in image enhancement and image restoration. - Apply various image processing techniques to solve real world problems.				
Unit No.	Course Content				Hours	
UNIT-I	Introduction and Digital Image Fundamentals: Digital Image Fundamentals, Elements of Digital Image Processing Systems, image as a 2D data, Image representation – Gray scale and Color images, image sampling and quantization, – Relationships between pixels.				10	
UNIT-II	Image enhancement in Spatial domain: Basic gray level Transformations, Histogram Processing Techniques, Histogram equalization, Histogram Matching, Spatial Filtering, Low pass filtering, High pass filtering, Image Enhancement (Frequency Domain): Filtering in Frequency Domain, Obtaining Frequency Domain Filters from Spatial Filters, Generating Filters Directly in the Frequency Domain, Low Pass (Smoothing) and High Pass (Sharpening) Filters in Frequency Domain.				15	
UNIT-III	Introduction to the Fourier transform and frequency domain concepts, Extension to functions of two variables, low pass filter, high pass filter, Laplace transformation, Image Smoothing, Image Sharpening, Homomorphic filtering.				10	
UNIT-IV	Image Restoration and Reconstruction: Various noise models, image restoration using spatial domain filtering, image restoration using frequency domain filtering, Estimating the degradation function, Inverse filtering.				10	
UNIT-V	Image Compression: Fundamentals of redundancies, Basic Compression Methods: Huffman coding, Arithmetic coding, Error free compression, Lossy compression. LZW coding, JPEG Compression standard. Image Segmentation: Detection of Discontinuities, Edge linking and boundary detection, Thresholding, Region Oriented Segmentation, Motion based segmentation.				15	

Recommended Learning Resources

Text Books:

1. Rafael C. Gonzalez, Richard E. Woods ,”Digital Image Processing (3rd Edition)”, Pearson, 2008
2. S Jayaraman, S Esakkirajan, T Veerakumar ,”Digital Image Processing”, TMH, 2010.

Reference Books:

1. Scotte Umbaugh, “Digital Image Processing and Analysis-Human and Computer Vision Application with using CVIP Tools (2nd Edition)”, CRC Press, 2011
2. Rafael C. Gonzalez, Richard E Woods and Steven L. Eddings, “Digital Image Processing using MATLAB (2nd Edition)”, TMH, 2010.
3. A.K.Jain, “ Fundamentals of Digital Image Processing” , PHI, 1989
4. Somka, Hlavac, Boyle “Digital Image Processing and Computer Vision”, Cengage Learning (Indian edition) 2008.
5. John C. Russ, J. Christian Russ, “Introduction to Image Processing & Analysis -2010“, CRC Press, 2010.

E-Resources:

Online Courses & Tutorials

- Alison’s “*Fundamentals of Digital Image Processing*”: a free-online course introducing DIP basics like image formation, geometry, and applications. [Alison](#)
- Coursera’s “*Fundamentals of Digital Image and Video Processing*” (by Northwestern University): covers spatial and frequency domain, enhancement, compression etc. [Coursera+1](#)
- Wolfram Research U’s “*Introduction to Image Processing*”: interactive, hands-on course using the Wolfram Language. [wolfram.com+1](#)
- GeeksforGeeks’ tutorial on DIP: great for quick reading, conceptual clarity, differences between types of images, operations etc. [GeeksforGeeks](#)
- A free 12-week course by SWAYAM (India) on Digital Image Processing, taught by Prof. Prabir K. Biswas. [Get Your Education](#)

Recommended e-Books

- Digital Image Processing by William K. Pratt – classic textbook for students/practitioners. [OverDrive](#)
- The Essential Guide to Image Processing by Alan C. Bovik – covers image-quality, watermarking, recognition etc. [OverDrive](#)
- Digital Image Processing Methods by Edward R. Dougherty – advanced methods and architectures. [OverDrive](#)
- Digital Image Processing and Analysis by Scott E. Umbaugh – engineering-approach + many exercises. [OverDrive](#)

How to Use These Resources

- **Start with tutorials/course** (like GeeksforGeeks or Alison) if you are beginner — get familiar with concepts: pixels, sampling, quantization, filters, histogram equalization.
- **Then progress into a textbook** (Pratt, Bovik, etc) for deeper theory, mathematics, algorithms.
- **Use interactive/hands-on courses** (Wolfram, SWAYAM) to code up these methods (MATLAB, Python, Wolfram language) and apply them.
- **Pick a specialisation:** image restoration, segmentation, pattern recognition, computer vision — then use the advanced books to deepen your knowledge.

Year	II	Course Code : SEPBCAT18		Credits	4
Semester	IV	Course Title : Data Science		Hours	60
Course Pre-requisites, if any		NA			
Formative Assessment Marks : 20		Summative Assessment Marks : 80		Duration of SEE: 3 hrs.	
Course Objectives		After completing this course satisfactorily, a student will be able to: - Understand the concepts of data and pre-processing of data. - Know simple pattern recognition methods. - Understand the basic concepts of Clustering and Classification. - Know the recent trends in Data Science.			
Unit No.	Course Content			Hours	
UNIT-I	Data Mining: Introduction, Data Mining Definitions, Knowledge Discovery in Databases(KDD) Vs Data Mining, DBMS Vs Data Mining, DM Techniques, Problems, Issues and Challenges in DM, DM applications.			15	
UNIT-II	Data Warehouse: Introduction, Definition, Multidimensional Data Model, Data Cleaning, Data Integration, and Transformation, Data reduction, Discretization.			10	
UNIT-III	Mining Frequent Patterns: Basic Concepts-Frequent Item Set Mining, Methods- Apriori and Frequent Pattern Growth(FPGrowth) algorithms-Mining Association Rules.			10	
UNIT-IV	Classification: Basic Concepts, Issues, Algorithms: Decision Tree Induction, Bayes Classification Methods, Rule-Based Classification, Lazy Learners(or Learning from your Neighbours), K Nearest Neighbour, Predication-Accuracy-Precision and Recall.			15	
UNIT-V	Clustering: Cluster Analysis, Partitioning Methods, Hierarchical Method, Density-Based Methods, Grid-Based Methods, Evaluation of Clustering.			10	
Recommended Learning Resources					
Reference Books:					
1. Arun K Pujari,” Data Mining Techniques (4 th Edition)”, Universities Press. 2. K.P. Soman, Shyam Diwakar,V. Ajay, “Data Mining-Theory and Practice”, PHI Publications.					

E-Resources:

Free Online Courses

- Alison – “*Introduction to Data Science*”: A completely free online course covering fundamentals of data science: processes, machine-learning intro, structured/unstructured data. [Alison+1](#)
- Alison – “*Data Science Masterclass for Beginners*”: A beginner-friendly course introducing skills & tools used in the field. [Alison](#)
- Alison – “*Diploma in Using Python for Data Science*”: Free course for learners with some Python knowledge aiming to apply it in data-science. [Alison](#)
- DataCamp – Interactive browser-based platform: lots of hands-on coding, practice in Python for data-science tasks. [DataCamp](#)
- Coursera – Course “*Code Free Data Science*”: Beginner-level, no programming required, focuses on data-science workflows. [Coursera](#)

e-Books & Reading Resources

- “*Data Science: Concepts and Practice*” by Vijay Kotu: Covers many commonly used data-science algorithms and methods for business/analytics practitioners. [OverDrive](#)
- “*Data Science*” by C. Raju: An accessible introduction to data science, with real-world examples, suitable for building foundations. [OverDrive](#)

How to Use These Resources

- **Start with a free course** — Use one of the Alison introductory courses to grasp the overall field and decide if you want to go deeper.
- **Choose a tool-based learning path** — If you prefer Python, jump into DataCamp and the Alison “Python for Data Science” course. If you prefer a non-coding starting path, try “Code Free Data Science” on Coursera.
- **Supplement with reading** — Use one of the e-books to deepen your understanding of algorithms and data-science processes.
- **Apply by doing projects** — Try small projects: e.g., analyze a dataset using Python/pandas, build simple visualizations, try a machine-learning model. This solidifies learning.
- **Certification** — If you need credentials, check whether the course offers a certificate (some free-courses do for optional fee) or use the skills to build a portfolio instead.

Year	III	Course Code : SEPBCAP16	Credits	2
Semester	VI	Course Title : PHP and MySQL Lab	Hours	50
Course Pre-requisites, if any		Knowledge of PHP Programming		
Formative Assessment Marks : 10		Summative Assessment Marks : 40	Duration of SEE: 3 hrs.	

PART - A:

1. Write a PHP script to print "hello world".
2. Write a PHP script to find odd or even number from given number.
3. Write a PHP script to find maximum of three numbers.
4. Write a PHP script to swap two numbers.
5. Write a PHP script to find factorial of a number.
6. Write a PHP script to check whether given number is palindrome or not
7. Write a PHP script to reverse s given number and calculate its sum
8. Write a PHP script to generate a Fibonacci series using Recursion function.
9. Write a PHP script to implement at least seven string functions.
10. Write a PHP script to insert new item in array on any positions in PHP.

PART - B:

1. Write a PHP script to implement constructor and destructor.
2. Write a PHP script to implement form handling suing get method.
3. Write a PHP script to implement form handling using post method.
4. Write a PHP script that receive form input by the method post to check the number is prime or not
5. Write a PHP script that receives string as a form input.
6. Write a PHP script to compute addition of two matrices as a form input.
7. Write a PHP script to show the functionality of date and time function.
8. Write a PHP script to upload a file.
9. Write a PHP script to implement database creation.
10. Write a PHP script to create table.
11. Write a PHP script to design a college admission form using MySQL database.

Instructions:

1. A certified journal is mandatory for every student to appear for the practical examination.
2. The student has to execute a minimum of eight programs from each part to complete the lab course.

Year	III	Course Code : SEPBCAP17	Credits	2
Semester	VI	Course Title: Digital Image Processing Lab	Hours	50
Course Pre-requisites, if any		Knowledge of SCI Lab/ MAT Lab		
Formative Assessment Marks : 10		Summative Assessment Marks: 40	Duration of SEE : 3 hrs.	

Note: Implement the following program by using SCI Lab/MAT Lab/Python Programming.

1. Write a program to read and display an image.
2. Write a program to convert RGB to gray scale image.
3. Write a program to crop an image.
4. Write a program to resize an image.
5. Write a program to display a negative image.
6. Write program to fill the region of interest for the image.
7. Write a program for implementation of relationship between pixels.
8. Write program for computation of Mean, Standard Deviation and Correlation coefficient of the given image.
9. Write program to obtain histogram equalization concept.
10. Write program to evaluate Salt and Pepper noise.
11. Write a program to evaluate Gaussian noise.
12. Write program for implementation of Image Smoothing Filters(Mean and Median Filtering of an image)
13. Write program for canny edge detection algorithm
14. Write a program for Robert edge detection method.
15. Write program for Sobel edge detection method.

Instructions:

1. A certified journal is mandatory for every student to appear for the practical examination.

Year	III	Course Code : Skill-3MP	Credits	6
Semester	VI	Course Title :Internship/Project	Hours	80
Formative Assessment Marks : 30		Summative Assessment Marks : 120 (Dissertation Evaluation:90 Marks + Viva-Voce:30)	Duration of SEE: 3 hrs.	

Course Outcomes (COs):

After completing the Project/Internship, students will be able to:

CO1: Apply the knowledge and skills acquired throughout the program to solve real-world or industry-related problems.

CO2: Design, develops, and implement a project or perform assigned tasks effectively in a professional environment.

CO3: Demonstrate the ability to work independently and collaboratively in teams to achieve project or organizational objectives.

CO4: Analyze and interpret data or project outcomes to provide feasible and sustainable solutions.

CO5: Communicate effectively through technical reports, presentations, and documentation.

CO6: Exhibit professional ethics, responsibility, and adaptability in a workplace or project setting.

CO7: Engage in self-learning and continuous improvement through reflection on internship/project experiences.

Execution of Project:

- The individual student is required to carry out the project under the guidance of course teacher.
- Project work problem statements shall be identified by the students with the help of course teachers and students shall submit the synopsis/project proposal of the same.
- During project development students are expected to define a project problem do requirements analysis, systems design, software development, apply testing strategies and do documentation with an overall emphasis on the development of a robust, efficient and reliable software systems.
- The project development process has to be consistent and should follow standards identified by the guide monitoring the project work.
- There is no restriction on use of hardware's and software's for carrying out the project work except that ready application packages are not allowed.
- The students have to submit two hard copies of Dissertation along with CD after completion of the project work.

Note:

1. Internship shall be Discipline Specific of 6 credits with duration 16 weeks.
2. The student should submit the final internship/Project report.
3. Method of evaluation: Presentations/Report submission/Activity/Viva-Voce.
4. The detailed guidelines and formats shall be formulated by the university separately.

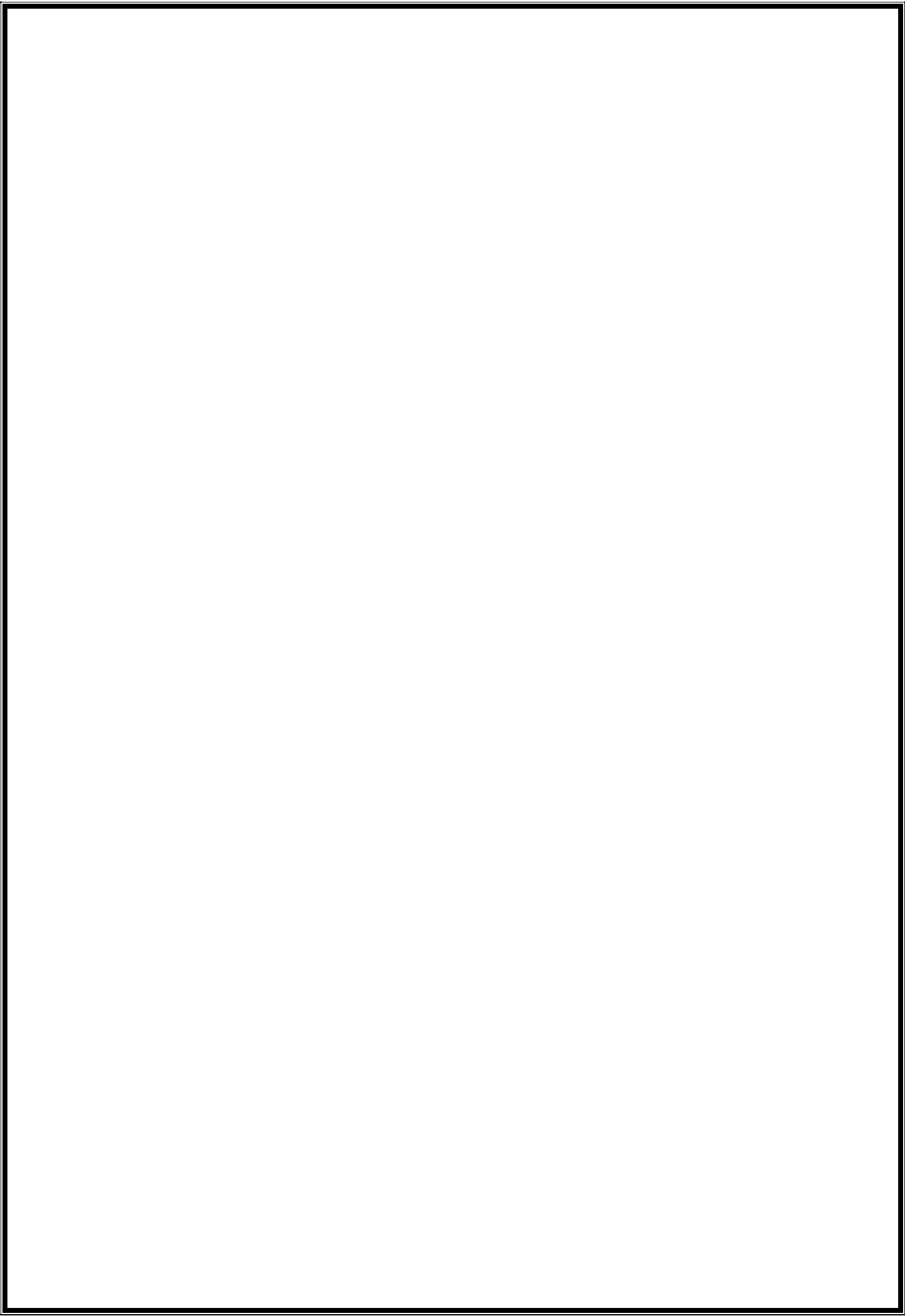
1	Summative Assessment	90 Marks
	Dissertation/Project Report evaluation. Presentation/Demo of the application developed (Navigation of the application, features incorporated, data validation, User Interface, Reports, etc.)	
2	Viva -Voce	30 Marks
3	Formative Assessment	30 Marks
4	Total	150 Marks

Theory Examination Question Paper Pattern for Major Subjects of B. Sc. / BCA**Time: 3 Hours****Max Marks: 80**

Instructions to Candidates:

1. All Sections are Compulsory.
2. Draw neat and labeled diagrams wherever necessary.

Section-A		
Answer ALL the following Questions.		10x2=20
Q1	a	Question to be asked from Unit-I
	b	Question to be asked from Unit-I
	c	Question to be asked from Unit-II
	d	Question to be asked from Unit-II
	e	Question to be asked from Unit-III
	f	Question to be asked from Unit-III
	g	Question to be asked from Unit-IV
	h	Question to be asked from Unit-IV
	i	Question to be asked from Unit-V
	j	Question to be asked from Unit-V
Section-B		
Answer any SIX of the following Questions.		6x5=30
Q2	Question to be asked from Unit-I	
Q3	Question to be asked from Unit-I	
Q4	Question to be asked from Unit-II	
Q5	Question to be asked from Unit-II	
Q6	Question to be asked from Unit-III	
Q7	Question to be asked from Unit-III	
Q8	Question to be asked from Unit-IV	
Q9	Question to be asked from Unit-V	
Section-C		
Answer any THREE of the following Questions.		3x10=30
Q10	Question to be asked from Unit-I	
Q11	Question to be asked from Unit-II	
Q12	Question to be asked from Unit-III	
Q13	Question to be asked from Unit-IV	
Q14	Question to be asked from Unit-V	



Instructions:

Continuous Internal Assessment for Practical	
Assessment Type	Marks
Test1 on PART A	10
Test2 on PART B	10
Maximum Marks to be awarded	10

Instructions:

1. A Certified Journal is mandatory for appearing the examination.
2. Students shall be given two programming assignments taking in to consideration of duration of the time allotted to students for writing, typing and executing the programs.
3. Award maximum 10 marks based on the average of two practical CIA.

Note: Guidelines given by the university from time-to-time shall be followed for I. A.

Semester End Examination (SEE) Scheme of Evaluation for Lab Examination

Assessment Criteria	Marks
Writing of 2 Programs(Each from Part A & Part B)	15
Execution (Includes program code modification and execution result)	15
Journal	05
Viva-Voice	05
Total	40

Instructions:

CIA(Continuous Internal Assessment) for Theory	
Assessment Type	Marks
1. Test 1	15
2. Assignment	05
1. Test 2	15
2. Seminar	05
Maximum Marks to be awarded	20

Award maximum 20 marks based on the average of two theories CIA.

Theory Examination Question Paper Pattern for Elective/Skill Papers of B. Sc. / BCA**Time: 2 Hours****Max Marks: 40**

Instructions to Candidates:

3. All Sections are Compulsory.
4. Draw neat and labeled diagrams wherever necessary.

Section-A		
Answer ALL the following Questions.		5x2=10
Q1	a	Question to be asked from Unit-I
	b	Question to be asked from Unit-I
	c	Question to be asked from Unit-I
	d	Question to be asked from Unit-II
	e	Question to be asked from Unit-II
Section-B		
Answer any SIX of the following Questions.		6x5=30
Q2		Question to be asked from Unit-I
Q3		Question to be asked from Unit-I
Q4		Question to be asked from Unit-I
Q5		Question to be asked from Unit-I
Q6		Question to be asked from Unit-II
Q7		Question to be asked from Unit-II
Q8		Question to be asked from Unit-II
Q9		Question to be asked from Unit-II

Instructions:

CIA(Continuous Internal Assessment) for Theory	
Assessment Type	Marks
1.Test 1	10
2.Test 2	10
Maximum Marks to be awarded	10

Award maximum 10 marks based on the average of two theories CIA.