

Online circulation of the Undergraduate Board of studies (UG BOS) held on 20th May, 2026.

Chaudhary Bansi Lal University, Bhiwani, Haryana

Scheme of Examination and Syllabus for Undergraduate Programme

Course: Computer Applications (Bachelor of Computer Applications AI/ML) (Under Multiple Entry-Exit, Internship and CBCS-LOCF in accordance to NEP-2020) w.e.f. 2025-26 in Phased Manner

Semester	Discipline-Specific Courses (DSC)	Paper Code	Nomenclature of Paper	Credit	Minor (MIC) / Vocational (VOC)	Credit	Multidisciplinary Courses (MDC)	Ability Enhancement Courses (AEC)	Skill Enhancement Courses (SEC) / Internship/ Dissertation	Value - Added Courses (VAC)	Total Credits
III	CC	25BCAIBM- N-301	Java Programming	3+1	25BCAIBM-N-304 Data Analysis using Excel	2+1	24UN-BBA-MDC-301 Fundamentals of Indian Capital Market@ 3 Credit	24UN-ENG-AEC301 English Grammar and Usage@2 Credit	25BCAIBM- N - SEC -301 Internship@4 Credit	24UN-GEO-VAC303 Culture of India @ 2 Credit	25
	CC	25BCAIBM- N-302	Operating system	3							
	CC	25BCAIBM- N-303	Discrete Mathematics	4							
IV	CC	25BCAIBM- N-401	Data Structure	3+1	25BCAIBM- N-404 Cloud Computing	3		24UN-ENG-AEC401 Vocabulary Building@ 2 Credit			17
	CC	25BCAIBM- N-402	Linux and Shell Programming	3+1							
	CC	25BCAIBM- N-403	Back End Engineering	3+1							
V	CC	25BCAIBM- N-501	Programming in Python	3+1	25BCAIBM-N-504 Web Security	3			25BCAIBM- N - SEC -501 Internship@4 Credit		19
	CC	25BCAIBM- N-502	Computer Networks	4							
	CC	25BCAIBM- N-503	Artificial Intelligence	4							
VI	CC	25BCAIBM- N-601	Internship	20							20

Prashant

Scheme of Examination for BCA(AI/ML)

Sr. No.	Course/ Paper Code	Title of the Course	Mode of Learning	Course Type	Credit			Contact Hours per Week			Examination Scheme				Total Marks
					Theory	Practical/ Seminar	Total	Theory	Practical /Seminar	Total	End Semester Exam	Internal Assessment	Practical (External)	Practical (Internal)	
1	25BCA/ML- N-301	Java Programming	CL/BL	CC	3	1	4	3	2	5	50	20	20	10	100
2	25BCA/ML- N-302	Operating system	CL/BL	CC	3	-	3	3	..	3	50	20	..	..	70
3	25BCA/ML- N-303	Discrete Mathematics	CL/BL	CC	4	..	4	4	..	4	70	30	..	..	100
4	25BCA/ML- N-304	Data Analysis using Excel	CL/BL	MIC	2	1	3	2	2	4	35	15	15	5	70
5	24UN-ENG- AEC301	English Grammar and Usage	CL/BL	AEC	2	..	2	2	..	2	35	15	..	..	50
6	25BCA/ML- N- SEC301	Internship	CL	SEC	..	4	4	..	..	..	..	..	70	30	100
10	24UN-BBA- MDC301	Fundamentals of Indian Capital Markets	CL	MDC	3	..	3	3	..	3	50	20	..	..	70
11	24UN-GEO- VAC303	Culture of India	CL	VAC	2	..	2	2	..	2	35	15	..	..	50
	Total						25			23					610

CC - Core Course; SEC-Skill Enhancement Course; AEC: Ability Enhancement Course MDC: Multidisciplinary Course VAC: Value Added course

MIC: Minor course

25BCAIML- N- 301	Java Programming	3L:0T:0P	3 Credits
------------------	------------------	----------	-----------

Maximum Marks:70
External Examination: 50
Internal Assessment: 20
Max. Time: 3 Hrs

Course Objectives

- To introduce the object-oriented programming concepts.
- To understand object-oriented programming concepts, and apply them in solving problems.
- To introduce the principles of inheritance and polymorphism; and demonstrate how they relate to the design of abstract classes
- To introduce the implementation of packages and interfaces
- To introduce the concepts of exception handling and multithreading.
- To introduce the design of Graphical User Interface using applets.

Note: There shall be nine questions in all. Question no. 1 shall be compulsory, consisting of eight short answer type questions covering the entire syllabus. Two questions will be asked from each unit. Students will have to attempt one question from each unit. Each question shall carry equal marks.

Unit I

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

Unit II

Interfaces: Interface basics; Defining, implementing and extending interfaces; Implementing multiple inheritance using interfaces. Packages: Basics of packages, Creating and accessing packages, System packages, Creating user defined packages

Unit III

Exception handling using the main keywords of exception handling: try, catch, throw, throws and finally; Nested try, multiple catch statements, creating user defined exceptions. File Handling Byte Stream, Character Stream, File I/O Basics, File Operations

Unit IV

AWT and Event Handling: The AWT class hierarchy, Events, Event sources, Event classes, Event Listeners, Relationship between Event sources and Listeners, Delegation event model, Creating GUI applications using AWT.

Course Outcomes

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

Forshen

- Able to design GUI based applications consisting of multiple classes.

Suggested Readings:

1. Balagurusamy, E., "Programming with JAVA, Tata McGraw Hill.
2. The Complete Reference JAVA, TMH Publication.
3. Beginning JAVA, Ivor Horton, WROX Public.
4. JAVA 2 UNLEASHED, Tech Media Publications.
5. Patrick Naughton and Herbert Schildt, "Java-2 The Complete Reference", TMH.

Forzhent

25BCAIML- N- 302	Operating system	3L:0T:0P	3 Credits
------------------	------------------	----------	-----------

Maximum Marks:70
External Examination: 50
Internal Assessment: 20
Max. Time: 3 Hrs

Course Objectives

- To understand the basic concepts, functions and types of operating systems.
- To understand Processes and Process scheduling
- To understand the concept of Deadlocks
- To analyze various memory management schemes.
- To understand I/O management and File systems.

Note: There shall be nine questions in all. Question no. 1 shall be compulsory, consisting of eight short answer type questions covering the entire syllabus. Two questions will be asked from each unit. Students will have to attempt one question from each unit. Each question shall carry equal marks.

Unit I

Introduction: Introduction to Operating System Concepts (Including Multitasking, Multiprogramming, Multi User, Multithreading etc.), Types of Operating Systems: Batch Operating System, Time-Sharing System, Distributed Operating System, Network Operating System, Real Time Operating System, Various Operating System services, architecture, System Programs and Calls.

Unit II

Process Management: Process Concept, Process Scheduling, Operations on Processes; CPU Scheduling, Scheduling Criteria, Scheduling Algorithms- First come First Serve (FCFS), Shortest- Job- First (SJF), Priority Scheduling, Round Robin (RR).
Deadlock: Methods for handling deadlock- Deadlock prevention, Avoidance

Unit III

Memory Management: Logical & Physical Address Space, Swapping, Contiguous Memory allocation, non-contiguous memory allocation, paging and segmentation techniques; Virtual Memory Management- Demand Paging & Page Replacement Algorithm; Demand Segmentation.

Unit IV

File System: Different types of files and their access methods, directory structures, various allocation methods, disk scheduling and management and its associated algorithms.

Course Outcomes

- Analyze various scheduling algorithms.
- Understand deadlock, prevention and avoidance algorithms.
- Compare and contrast various memory management schemes.
- Understand the functionality of file systems

Suggested Readings:

For more

1. Operating System Concepts by Silberschatz et al 5th Edition 1998, Addison-Wesley.
2. Modern Operating System by A. Tanenbaum, 1992, Prentice Hall
3. Operating System By Peterson, 1985, AW.
4. Operating System By Milankovik, 1990, THM
5. Operating System Incorporating with Unix & Windows by Colin Ritchie, 1974, THM.

Forshen

25BCAIML- N- 303	Discrete Mathematics	4L:0T:0P	4 Credits
------------------	----------------------	----------	-----------

Maximum Marks: 100
External Examination: 70
Internal Assessment: 30
Max. Time: 3 Hrs

Course Objectives

- To provide students with a solid foundation in set theory, logic, inference, counting, relations, graph theory, algebraic structures, and computability theory, equipping them with the skills to solve complex mathematical problems and apply these concepts in computer science.

Note: There shall be nine questions in all. Question no. 1 shall be compulsory, consisting of eight short answer type questions covering the entire syllabus. Two questions will be asked from each unit. Students will have to attempt one question from each unit. Each question shall carry equal marks.

Unit I

Sets & Logic: Sets - Visualizing sets, understanding membership, operations on sets (union, intersection, difference, complement). Algebra of sets and its laws, Mathematical Induction, Relations, Types of Relation- reflexive, symmetric, antisymmetric, Equivalence Relations
Logic - Introduction to propositional logic, truth tables, Logical Operations, Conditional Statements

Unit II

Vectors & Matrix: Vectors - Operations, Column Vectors, Matrices - Matrix Addition, Matrix Multiplication, Transpose, Square Matrices and Algebra on Square Matrices, Inverse of Square Matrix
Functions & Algorithms: Functions as Relations, One-to-One, Onto and Invertible Functions, Mathematical, Exponential and Logarithmic Functions, Recursively Defined Functions (Examples like- Factorial, Fibonacci), Algorithm and Functions - Polynomial Evaluation (Direct Method, Synthetic Division), Complexity of Algorithm - Best Case, Worst Case, Average Case

Unit III

Counting Techniques: Counting Principles - Sum Rule and Product Rule. Mathematical Functions - Factorial Function, Binomial Coefficients. Binomial Theorem - Binomial Coefficients and Pascal's Triangle. Permutations & Combinations - Understanding arrangements (permutations) and selections (combinations), using formulas for common scenarios. Tree Diagram

Unit IV

Graph - Graphs and Multigraphs, Degree of Vertex, Subgraphs, Isomorphic Graphs, Homeomorphic Graphs, Path, Connected Graph, Labeled Graph, Weighted Graph, Directed Graph, Undirected Graph, Complete Graphs, Regular Graphs, Bipartite Graphs, Planar Graph, Maps & Regions - Euler's Formula, Representing Graphs in Memory - Adjacency Matrix, Linked Representation of a Graph
Tree - Tree Graphs, Spanning Trees, Minimum Spanning Trees, Binary Trees, Representing Binary Trees in Memory - Linked & Sequential, Traversing Binary Tree, Binary Search Tree

Prashant

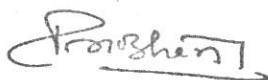
Course Outcomes:

A student will be able to:

- Understand sets, sequences, matrices, and logical operations..
- Use statements, connectives, and normal forms in logic.
- Apply inference theories in statement and predicate calculus.
- Solve problems using permutations, combinations, and recurrence relations; manipulate relations and digraphs
- Understand graph theory concepts, Boolean algebras, and basic computability; explore semi-groups and groups.

Suggested Readings:

1. E. Goodaire , "Discrete Mathematics with Graph theory", PHI,. ISBN--10: 0131679953 2) J. K. Sharma, "Discrete Mathematics", McMillan, ISBN-9780230322301
2. J.P. Tremblay and R. Manohar, "Discrete Mathematical Structure with Applications to Computer Science", TMH, ISBN- 0-07-463113-6
3. Bernard Kolman, Robert C. Busby and Sharon Ross, "Discrete Mathematical Structure", PHI, ISBN- 978-81-203-3689-6
4. C. L. Liu, "Elements of Discrete Mathematics", TMH



25BCAIML-N-304	Data Analysis using Excel	3L:0T:0P	2 Credits
----------------	----------------------------------	----------	-----------

Maximum Marks:50
External Examination: 35
Internal Assessment: 15
Max. Time: 3 Hrs

Course Objectives

- Understand the fundamental elements of a dashboard, including charts, tables, and visuals, to effectively present data-driven insights.
- Develop skills in utilizing advanced data analysis tools to derive actionable insights and drive data-informed decision-making processes.
- Explore methods for linking Excel with databases, facilitating seamless data exchange between Excel and external data sources such as Access and SQL Server.

Note: There shall be nine questions in all. Question no. 1 shall be compulsory, consisting of eight short answer type questions covering the entire syllabus. Two questions will be asked from each unit. Students will have to attempt one question from each unit. Each question shall carry equal marks.

Unit I

Introduction to Excel for Data Analysis-Overview of Excel interface, Basics of spreadsheets, Data types and formats, Basic formulas and functions. Data Entry and Cleaning- Data entry techniques, importing data from various sources (CSV, Text, Web), Data cleaning techniques, removing duplicates, handling missing values.

Unit II

Basic Data Manipulation- Sorting and filtering data, using tables in Excel, Conditional formatting, Data validation, Advanced Formulas and Functions- Logical functions: IF, AND, OR; Lookup functions: VLOOKUP, HLOOKUP, INDEX, MATCH; Text functions: CONCATENATE, LEFT, RIGHT, MID, TRIM, Date and time functions

Unit III

Data Analysis Techniques-Descriptive statistics: mean, median, mode, standard deviation, Using the Analysis Toolpack, PivotTables and PivotCharts, Grouping and summarizing data
Data Visualization- Creating and formatting charts: bar, line, pie, scatter, Advanced charting techniques, Using Sparklines, Dynamic charts with interactive controls.

Unit IV

Introduction to Dashboards- Elements of a dashboard, designing interactive dashboards, Using form controls and slicers, Linking charts and tables for dynamic updates.
Advanced Data Analysis Tools- What-If analysis: Scenario Manager, Goal Seek, Data Tables, Solver add-in for optimization problems, Using Power Query for data extraction and transformation, Integrating Excel with Other Tools-Linking Excel with databases, Exporting and importing data with other software (e.g., Access, SQL Server), Using Excel with cloud services (e.g., Google Sheets, Power BI).

Prashant

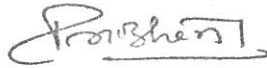
Course Outcomes

A student will be able to

- Design and create interactive dashboards that effectively communicate complex data insights to stakeholders.
- Apply techniques like What-If analysis, optimization, and data transformation tasks using advanced Excel tools.
- Demonstrate the ability to import, export, and manipulate data across different platforms, improving data interoperability and workflow efficiency.
- Apply their understanding of dashboard design, advanced data analysis, and data integration to solve complex business challenges effectively.

Suggested Readings:

1. "Information Dashboard Design: Displaying Data for At-a-Glance Monitoring" by Stephen Few
2. "Excel 2019 Bible" by Michael Alexander and Richard Kusleika
3. "Power Pivot and Power BI: The Excel User's Guide to DAX, Power Query, Power BI & Power Pivot in Excel 2010-2016" by Rob Collie and Avichal Singh
4. "SQL Server 2019 Integration Services Cookbook" by Christian Cote, Matija Lah, Dejan Sarka, and David Dye.



Java Programming Lab (Based on 25BCAIML- N-301)	0L:0T:2P	1 Credits
---	----------	-----------

Maximum Marks:30
External Examination: 20
Internal Assessment: 10

Lab 1: Basics

- Write a Java program that prints "Hello, World!" to the console.
- Create a Java program that calculates the factorial of a given number using a loop.
- Develop a Java program to print the Fibonacci series up to a given limit.

Lab 2: Interface

- WAP to find the area of rectangle and circle using Interface.

Lab 3: OOPS

Create a class Film with string objects which stores name, language and lead_actor and category (action/drama/fiction/comedy). Also include an integer data member that stores the duration of the film. Include parameterized constructor, default constructor and accessory functions to film class. Film objects can be initialized either using a constructor or accessor functions. Create a class FilmMain that includes a main function. In the main function create a vector object that stores the information about the film as objects. Use the suitable methods of vector class to iterate the vector object to display the following • The English film(s) that has Arnold as its lead actor and that runs for shortest duration. • The Tamil film(s) with Rajini as lead actor. • All the comedy movies.

Lab 4: OOPS inheritance

Define an abstract class 'Themepark' and inherit 2 classes 'Queensland' and 'Wonderland' from the abstract class. In both the theme parks, the entrance fee for adults is Rs. 500 and for children it is Rs. 300. If a family buys 'n' adult tickets and 'm' children tickets, define a method in the abstract class to calculate the total cost. Also, declare an abstract method playGame() which must be redefined in the subclasses. In Queensland, there are a total of 30 games. Hence create a Boolean array named 'Games' of size 30 which initially stores false values for all the elements. If the player enters any game code that has already been played, a warning message should be displayed and the user should be asked for another choice. In Wonderland, there are a total of 40 different games. Thus create an integer array with 40 elements. Here, the games can be replayed, until the user wants to quit. Finally display the total count of games that were repeated and count of the games which were not played at all.

Lab 5: Inheritance

Define a superclass Shape with an abstract method calculateArea(). Implement subclasses Circle and Rectangle that inherit from Shape and override the calculateArea() method to calculate the area specific to each shape. Create objects of both subclasses and demonstrate polymorphism.

Lab 6: Exception Handling

Read the Register Number and Mobile Number of a student. If the Register Number does not contain exactly 9 characters or if the Mobile Number does not contain exactly 10

Prashant

characters, throw an `IllegalArgumentException`. If the Mobile Number contains any character other than a digit, raise a `NumberFormatException`. If the Register Number contains any character other than digits and alphabets, throw a `NoSuchElementException`. If they are valid, print the message 'valid' else 'invalid'.

Lab 7: Package

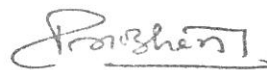
Write a program to demonstrate the knowledge of students in working with user-defined packages and sub-packages. Eg., Within the package named 'primespackage', define a class `Primes` which includes a method `checkForPrime()` for checking if the given number is prime or not. Define another class named `TwinPrimes` outside of this package which will display all the pairs of prime numbers whose difference is 2. (Eg, within the range 1 to 10, all possible twin prime numbers are (3,5), (5,7)). The `TwinPrimes` class should make use of the `checkForPrime()` method in the `Primes` class.

Lab 8: File Handling

Write a program to demonstrate the knowledge of students in File handling. Eg., Define a class 'Donor' to store the below mentioned details of a blood donor. Name, age, Address, Contactnumber, bloodgroup, date of last donation Create 'n' objects of this class for all the regular donors at Vellore. Write these objects to a file. Read these objects from the file and display only those donors' details whose blood group is 'A+ve' and had not donated for the recent six months.

Lab 9: AWT

- Write AWT program in Java to find the sum, Multiplication and average of three numbers entered in three Text fields by clicking the corresponding Labeled Button. The result should be appearing in fourth text field.
- Write Java Program to generate Even numbers and Odd Numbers in TextField "T1 and T2 respectively" while pressing Button "Even" and "Odd".



Data Analysis using Excel Lab (Based on 25BCAIML-N-304)	0L:0T:2P	1 Credits
--	----------	-----------

Maximum Marks:20
External Examination: 15
Internal Assessment: 5

Practical1: Data Entry and Data Cleaning

- **Task:** Create a sample dataset with 10 rows and 5 columns.
- **Steps:**
 1. Enter data into the spreadsheet.
 2. Use Excel's data cleaning tools to remove duplicates, handle missing values, and format data.

Practical 2: Sorting and Filtering Data

- **Task:** Sort and filter a dataset to analyze specific data subsets.
- **Steps:**
 1. Sort data by different columns.
 2. Apply filters to display specific data subsets based on criteria.

Practical 3: Using Formulas and Functions

- **Task:** Apply basic and conditional functions to a dataset.
- **Steps:**
 1. Use basic functions like SUM, AVERAGE, MIN, and MAX.
 2. Apply conditional functions such as IF, COUNTIF, and SUMIF.

Practical 4: Data Validation

- **Task:** Implement data validation rules to ensure accurate data entry.
- **Steps:**
 1. Create drop-down lists.
 2. Set validation rules for data entry to ensure data integrity.

Practical 5: Conditional Formatting

- **Task:** Use conditional formatting to highlight data based on specific criteria.
- **Steps:**
 1. Highlight cells based on specific criteria.
 2. Use color scales, data bars, and icon sets for visual representation.

Practical 6: Creating and Formatting Tables

- **Task:** Convert data ranges into tables and apply table styles.
- **Steps:**
 1. Convert a range of data into an Excel table.
 2. Apply different table styles and properties for better readability.

For 3 hours

Practical 7: Charts and Graphs

- **Task:** Visualize data by creating various types of charts.
- **Steps:**
 1. Create bar, line, and pie charts.
 2. Customize chart elements such as titles, labels, and legends.

Practical 8: Pivot Tables

- **Task:** Summarize and analyze large datasets using pivot tables.
- **Steps:**
 1. Create pivot tables from large datasets.
 2. Analyze data by rearranging fields in the pivot table.

Practical 9: Pivot Charts

- **Task:** Create and customize pivot charts to visualize pivot table data.
- **Steps:**
 1. Create pivot charts from pivot tables.
 2. Customize pivot chart elements for clarity.

Practical 10: VLOOKUP and HLOOKUP Functions

- **Task:** Retrieve data using VLOOKUP and HLOOKUP functions.
- **Steps:**
 1. Use VLOOKUP to find data in vertical columns.
 2. Use HLOOKUP to find data in horizontal rows.

Practical 11: Text Functions

- **Task:** Clean and manipulate text data using text functions.
- **Steps:**
 1. Use functions like LEFT, RIGHT, MID, and CONCATENATE.
 2. Clean and format text data as required.

Practical 12: Date and Time Functions

- **Task:** Perform calculations with date and time data.
- **Steps:**
 1. Use functions like TODAY, NOW, DATE, and TIME.
 2. Calculate differences between dates and times.

Prashant

Scheme of Examination for BCA (AI/ML)

Semester: 4th (w.e.f. Session 2025-26)

S r . N o .	Course/ Paper Code	Title of the Course	Mode of Learning	Course Type	Credit			Contact Hours per Week			Examination Scheme				Total Marks
					Theory	Practical / Seminar	Total	Theory	Practical /Seminar	Total	End Semester Exam	Internal Assessment	Practical (External)	Practical (Internal)	
1	25BCAIML-N-401	Data Structure	CL/B L	CC	3	1	4	3	2	5	50	20	20	10	100
2	25BCAIML-N-402	Linux and Shell Programming	CL/B L	CC	3	1	4	3	2	5	50	20	20	10	100
3	25BCAIML-N-403	Back End Engineering	CL/B L	CC	3	1	4	3	2	5	50	20	20	10	100
4	25BCAIML-N-404	Cloud Computing	CL/B L	MIC	3	..	3	3	..	3	50	20	..		70
5	24UN-ENG-AEC401	Vocabulary Building	CL/B L	AEC	2	..	2	2	..	2	35	15	..		50
	Total						17			20					420

CC - Core Course;

MIC: Minor course

AEC: Ability Enhancement Course

Pranishant

25BCA1ML- N- 401	Data Structure	3L:0T:0P	3 Credits
------------------	----------------	----------	-----------

Maximum Marks:70
External Examination: 50
Internal Assessment: 20
Max. Time: 3 Hrs

Course Objectives

- The Purpose of the course is to introduce the principles and paradigms of programming languages for design and implement the software intensive systems.

Note: There shall be nine questions in all. Question no. 1 shall be compulsory, consisting of eight short answer type questions covering the entire syllabus. Two questions will be asked from each unit. Students will have to attempt one question from each unit. Each question shall carry equal marks.

Unit I

Data Structure Definition, Data Type vs. Data Structure, Classification of Data Structures, Data Structure Operations, Applications of Data Structures.

Algorithm Specifications: Performance Analysis and Measurement (Time and Space Analysis of Algorithms- Average, Best and Worst Case Analysis).

Arrays: Introduction, Linear Arrays, Representation of Linear Array in Memory, Two Dimensional and Multidimensional Arrays, Sparse Matrix and its Representation, Operations on Array: Algorithm for Traversal, Selection, Insertion, Deletion and its implementation.

Unit II

String Handling: Storage of Strings, Operations on Strings viz.,Length, Concatenation, Substring, Insertion, Deletion, Replacement,Pattern Matching

Linked List: Introduction, Array vs. linked list, Representation of linked lists in Memory, Traversing a Linked List, Insertion, Deletion,Searching into a Linked list, Type of Linked List.

Unit III

Stack: Array Representation of Stack, Linked List Representation of Stack, Algorithms for Push and Pop, Application of Stack: Polish Notation, Postfix Evaluation Algorithms, Infix to Postfix Conversion, Infix to Prefix Conversion, Recursion.

Introduction to Queues: Simple Queue, Double Ended Queue, Circular Queue, Priority Queue, Representation of Queues as Linked List and Array, Applications of Queue. Algorithm on Insertion and Deletion in Simple Queue and Circular Queue. Priority Queues.

Unit IV

Tree: Definitions and Concepts, Representation of Binary Tree, Binary Tree Traversal (Inorder, postorder, preorder), Binary Search Trees – Definition, Operations viz., searching, insertions and deletion; Searching and Sorting Techniques, Sorting Techniques: Bubble sort, Merge sort, Selection sort, Quick sort, Insertion Sort. Searching Techniques: Sequential Searching, Binary Searching.

Course Outcomes

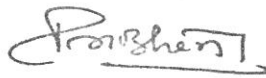
Pragathi

A student will be able to:

- Understand and analyze the basic concepts of complexity of an algorithm.
- Describe how arrays, linked lists, stacks, queues, trees, and graphs are represented in memory, used by the algorithms and their common applications.
- Design & Analyze the searching and sorting algorithms.
- Implement the Graphs and also perform various operations on this data structure.
- Understand & Implement the various concepts of stacks and queues

Suggested Readings:

1. Lipschutz, "Data Structures" Schaum's Outline Series, Tata McGraw-hill Education (India) Pvt. Ltd.
2. Thareja, "Data Structure Using C" Oxford Higher Education.
3. Aaron M. Tenenbaum, YedidyahLangsam and Moshe J. Augenstein, "Data Structures Using C and C++", PHI Learning Private Limited, Delhi India
4. Horowitz and Sahani, "Fundamentals of Data Structures", Galgotia Publications Pvt Ltd Delhi India.



25BCA1ML- N- 402	Linux and Shell Programming	3L:0T:0P	3 Credits
------------------	-----------------------------	----------	-----------

Maximum Marks:70
External Examination: 50
Internal Assessment: 20
Max. Time: 3 Hrs

Course Objectives

- The Purpose of the course is to introduce the Unix/Linux operating system and shell programming basics and knowledge of Unix/Linux tools for file handling.

Note: There shall be nine questions in all. Question no. 1 shall be compulsory, consisting of eight short answer type questions covering the entire syllabus. Two questions will be asked from each unit. Students will have to attempt one question from each unit. Each question shall carry equal marks.

Unit I

Introduction to Linux: Linux distributions, Overview of Linux operating system, Linux architecture, Features of Linux, Accessing Linux system, Starting and shutting down system. Logging in and Logging out, Comparison of Linux with other operating systems.

Unit II

Commands in Linux: General-Purpose commands, File oriented commands, directory oriented commands, Communication-oriented commands, process oriented commands, etc. Regular expressions & Filters in Linux: Simple filters viz. more, wc, diff, sort, uniq, grep; Introducing regular expressions.

Unit III

Linux file system: Linux files, inodes and structure and file system, file system components, standard file system, file system types. Processes in Linux: Starting and Stopping Processes, Initialization Processes, Mechanism of process creation, Job control in linux using at, batch, cron& time.

Unit IV

Shell Programming: vi editor, shell variables, I/O in shell, control structures, loops, subprograms, creating & executing shell scripts in linux.

Course Outcomes

A student will be able to:

- Understand the Unix/Linux operating system, installation procedures, basic utilities command and basics of shell programming.
- Organize & handle the files or records through various Unix/Linux tools by using regular expression.
- Analyze C/C++ environment, Makefile, libraries, linking, debugging methodologies and apply the advantages of Python programming.
- Assess the processes and manage the networks securely in Linux. CO5: Analyze process manipulation and learning password management for user privileges in Linux.

Prashant

Suggested Readings:

1. John Goerzen: Linux Programming Bible, IDG Books, New Delhi, 2000 Neil
2. Matthew, Richard Stones: Beginning Linux Programming, Wrox-Shroff, 2007

Goerzen

25BCAIML- N- 403	Back End Engineering	3L:0T:0P	3 Credits
------------------	----------------------	----------	-----------

Maximum Marks:70
 External Examination: 50
 Internal Assessment: 20
 Max. Time: 3 Hrs

Course Objectives

- The Purpose of the course is to introduce the concepts of Backend-Engineering.

Note: There shall be nine questions in all. Question no. 1 shall be compulsory, consisting of eight short answer type questions covering the entire syllabus. Two questions will be asked from each unit. Students will have to attempt one question from each unit. Each question shall carry equal marks.

Unit I

Client Server Architecture: Introduction to client Server Architecture, Understanding Server and how request is handled, Understanding Nodejs, advantages and disadvantages, Folder structure and basic Node server and understanding modules

Creating basic node server, handling request, creating endpoints, Modules, import modules Handling static pages with file stream, Handling exception, callback and other methods of endpoints,

Unit II

Express and Routing: Introduction to Express- Request lifecycle in express, how request travels in express, blocking vs non-blocking code, Handling static file.

Routing and different type like Static, Dynamic.

Unit III

Middleware and Client Server Management: Intro to Middlewares, Middleware lifecycle, serve static files, Body parser, Introduction to Multer Disk storage and File Filter handling in Multer Client state management Introduction, Client state management using session module, Different ways of managing session, Node style async error handling.

Authentication: Authentication, Authorization, Handling Exceptions during authorization And authentication

SSR - How to use EJS to template your Node.js application, Create the EJS Partial, Pass Data to Views and Partial

Unit IV

Database: Introduction to databases, SQL and No SQL Databases, MongoDB Shell and MongoDB Compass Installation, MongoDB Shell and commands, Applying CRUD operations on database and creating Pages using EJS

Course Outcomes

A student will be able to:

- Understand Backend technologies with Nodejs and MongoDB.
- Understand and Create Nodejs server, handling request, creating endpoints.

Prashant

- Understand and applying Session Management and Routing.
- Use of NoSQL database MongoDB and applying CRUD operations on MongoDB.
- Apply advanced Excel tools and features, such as PivotTables and Power Query, to handle complex data analysis tasks

Suggested Readings:

1. Eloquent JavaScript by Marijn Haverbeke, No Starch Press, 3rd edition (2018).
2. Mastering Node.js by Sandro Pasquali, Packt Publishing Ltd, Second Edition
3. MongoDB Basics by Peter Membrey, David Hows and Eelco Plugge, Apress, 1st Edition.

Parghen

25BCAIML- N- 404	Cloud Computing	3L:0T:0P	3 Credits
------------------	-----------------	----------	-----------

Maximum Marks:70
External Examination: 50
Internal Assessment: 20
Max. Time: 3 Hrs

Course Objectives

- The Purpose of the course is to understand the evolution of computer networks using the layered network architecture

Note: There shall be nine questions in all. Question no. 1 shall be compulsory, consisting of eight short answer type questions covering the entire syllabus. Two questions will be asked from each unit. Students will have to attempt one question from each unit. Each question shall carry equal marks.

Unit I

Cloud Computing – Definition of Cloud – Evolution of Cloud Computing – Underlying Principles of Parallel and Distributed, History of Cloud Computing - Cloud Architecture - Types of Clouds - Business models around Clouds – Major Players in Cloud Computing Issues in Clouds - Eucalyptus - Nimbus - Open Nebula, CloudSim.

Unit II

Types of Cloud services: Software as a Service Platform as a Service –Infrastructure as a Service - Database as a Service - Monitoring as a Service –Communication as services. Service providers- Google, Amazon, Microsoft Azure, IBM, Sales force, S3 Storage.

Case Study: Creating EC2 Instances of Windows and Linux in AWS, Creating and Accessing Relational Database Services in AWS.

Unit III

Email Communication over the Cloud - CRM Management – Project Management-Event Management - Task Management – Calendar - Schedules - Word Processing – Presentation – Spreadsheet - Databases – Desktop - Social Networks and Groupware.

Case Study: CRM Management: Setting up a Salesforce Account and customizing dashboards & reports in CRM Tools

Unit IV

Need for Virtualization – Pros and cons of Virtualization – Virtualization and Cloud - Types of Virtualizations –System VM, Process VM, Virtual Machine monitor – Virtual machine properties - Interpretation and binary translation, HLL VM - supervisors – Xen, KVM, VMware, Virtual Box, Hyper-V.

Case Study: Installing and configuring Hyper-V/ KVM for installation of a Guest Operating System.

Security in Clouds: Cloud security challenges – Software as a Service Security, Common Standards: The Open Cloud Consortium – The Distributed Management Task Force – Standards for Application Developers Standards for Messaging – Standards for Security, End user access to cloud computing, Mobile Internet devices and the cloud.

Parag Mehta

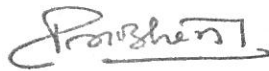
Course Outcomes

A student will be able to:

- Illustrate the evolution and issues in Cloud Computing.
- Demonstrate the different services of Cloud.
- Analyze different services of cloud in industry prospective.
- Show the concept of virtualization in Cloud computing.
- Illustrate the core issues of cloud computing such as resource management and security.

Suggested Readings:

1. Haley Beard, "Best Practices for Managing and Measuring Processes for On-demand Computing, Applications and Data Centers in the Cloud with SLAs", Emereo Pty Limited, July 2008.
2. G. J. Popek, R.P. Goldberg, "Formal requirements for virtualizable third generation Architectures, Communications of the ACM", No.7 Vol.17, July 1974
3. David E.Y. Sarna, "Implementing and Developing Cloud Application", CRC press 2011. 2. Lee Badger, Tim Grance, Robert Patt-Corner, Jeff Voas, NIST, Draft cloud computing synopsis and recommendation, May 2011.



Data Structure Lab (Based on 25BCAIML- N- 401	0L:0T:2P	1 Credits
---	----------	-----------

Maximum Marks:30
External Examination: 20
Internal Assessment: 10

1. Write a program to find the 2nd maximum element in an array.
2. Given a list of integers, write a program to find out the number that has the highest frequency. If there are more than one such numbers, output the smaller one.
3. Given two arrays of the same size, write a program to find out if given arrays are equal or not.
Note: Two arrays are said to be equal if both of them contain the same set of elements, although arrangements (or permutation) of elements may be different.
4. Given a string, write a program to test the characters for their uniqueness. If all the characters occur at most 1 time in the string, then print "YES", otherwise if some character occurs at least twice in the string print "NO".
5. Write a program to count the numbers of words in a string.
A word is defined as text separated by space(' ') or multiple spaces.
6. Write a program to find the smallest and the largest element of a given linked list and move the smallest node to the front and move the largest node to the end of the list.
7. Write a program to check whether it contains a loop or not. If there is a loop return the number of nodes in the loop, otherwise return 0.
8. Write a program to rotate a given doubly linked list counterclockwise by k nodes.
9. Write a program to insert in a sorted circular linked list.
10. Write a program to convert the infix to postfix expression.
11. Write a program to implement a queue using a linked list.
12. Given a binary tree, write a program to print all paths from root node to leaf nodes with their respective lengths and total number of paths in it.
13. Write a program to print a binary tree with level order traversal.
14. Write a program to find the kth smallest element in the binary search tree.
15. Given a sorted array of n elements, and an integer k. Write a program to find whether there exists any two elements in the array that sums up to the given value k. If such a pair is found, then print "Found", else print "Not Found".

Prashant

Linux and Shell Programming Lab (Based on 25BCA1ML- N-402)

0L:0T:2P

1 Credits

Maximum Marks:30
External Examination: 20
Internal Assessment: 10

Experiment	Experiment
1	Installation of Red Hat Linux using VMware Workstation or Oracle Virtual Box.
2	Demonstrate general purpose commands in Linux- cal, who, date, echo, bc, tty.
3	Demonstrate Directory and File Management in Linux- mkdir, rmdir, cd, pwd, ls, cat, rm, mv, cp.
4	Demonstrate to handle File and Directory permissions through: ls -l, ls -ld, chmod.
5	Show use of vi-editor and its various options.
6	Demonstrate the working of all process handling commands: ps, fg, bg, nice, renice, top etc.
7	Copy /etc/passwd file from your Linux system and demonstrate filters in Linux: more, less, diff, sort, Grep and sed.
8	Using crontab schedule below tasks in your Linux Server: ·Run a command at 15:00 on every day from Monday through Friday · Run a script every day, every hour, on the hour, from 8 AM through 4 PM · Run the a script at 9:15pm, on the 1st and 15th of every month · Set custom HOME, PATH, SHELL and MAILTO variables and run a command every minute

Prashant

9	Use at command to schedule: · To schedule a job for the coming Monday at a time twenty minutes later than the current time · To schedule a job to run at 1:45 on August 12, 2024 · schedule a job to run at 3 PM four days from the current time · schedule a job to shut down the system at 4:30 PM today
10	WAP that accepts user name and reports if a user is logged in or not. If logged in, display a greeting for him.
11	Create a shell script that checks if a file exists in the current directory or not.
12	WAP that takes a filename as input and checks if it is executable, if not make it executable
13	Create a Menu driven shell script which manages users and display following menu 1. create a new user 2. set passwords for new user or update passwords for existing users 3. Lock the user password 4. locate the home directory for the user and create its backup file with timestamp 5. View all the authentic users and their id
14	Write a shell script that calculates the sum of integers from 1 to N using a loop.
15	Create a script that searches for a specific word in a file and counts its occurrences.

Prashant

Back End Engineering LAB (Based on 25BCAIML-N-403)	0L:0T:2P	1 Credits
--	----------	-----------

Maximum Marks:30
 External Examination: 20
 Internal Assessment: 10

1. Nodejs Installation and Intro to Modules [user defined and built-in]
2. Creating a basic HTTP Server and returning different types of content in Node.js
3. Handling Filestream
4. Creating a Node server with Express
5. Handling Static Pages with Filestream
6. Handling Exception
7. Example to implement callback and other methods of endpoints
8. Express: creating server using Express
9. Static Routing using express
10. Dynamic Routing using express
11. static and body parser middleware
12. Client state management using session module
13. Handling Exceptions during auth. and authentication
14. Multer: uploading single file
15. Multer: uploading multiple file
16. EJS Server side rendering
17. Create the EJS Partial and Pass Data to Views and Partial
18. MongoDB command-line Interface creating database, collection, different operations on Database
19. Connecting mongodb with nodejs application using sign up pages
20. Applying CRUD operations on database
21. Login Management using Mongoose with ATLAS

Prashant

22. CRUD Operations using Mongoose with validation

23. Authentication using JSON using Web Tokens

Poozhen

Semester: 5th (w.e.f. Session 2025-26)

Scheme of Examination for BCA (AI/ML)

S r . N o .	Course/ Paper Code	Title of the Course	Mode of Learnin g	Course Type	Credit			Contact Hours per Week			Examination Scheme			Total Mark s
					Theo ry	Practical / Seminar	Total	Theo ry	Practi cal /Semi nar	Total	(End Semest er Exa m)	Int ernal Asse ssme nt	Prac tical (Inte mal)	
1	25BCA/ML- N- 501	Programming in Python	CL/BL	CC	3	1	4	3	2	5	50	20	10	100
2	25BCA/ML- N- 502	Computer Network	CL/BL	CC	4	-	4	4	..	4	70	30		100
3	25BCA/ML- N- 503	Artificial Intelligence	CL/BL	CC	4	-	4	4	..	4	70	30		100
4	25BCA/ML- N- 504	Web Security	CL	MIC	3	-	3	3	..	3	50	20		70
5	24BCA/ML- N SEC -501	Internship	CL	SEC	..	4	4	..		..	70	30		100
	Total						19			16				470

CC - Core Course; SEC-Skill Enhancement Course MIC-Minor
course

Pranishant

25BCAIML- N- 501	Programming in Python	3L:0T:0P	3 Credits
------------------	----------------------------------	----------	-----------

Maximum Marks:70
External Examination: 50
Internal Assessment: 20
Max. Time: 3 Hrs

Course Objectives

- To understand why Python is a useful scripting language for developers.
- To learn how to design and program Python applications.
- To learn how to use lists, tuples, and dictionaries in Python programs.
- To learn how to use exception handling in Python applications for error handling.

Note: There shall be nine questions in all. Question no. 1 shall be compulsory, consisting of eight short answer type questions covering the entire syllabus. Two questions will be asked from each unit. Students will have to attempt one question from each unit. Each question shall carry equal marks.

Unit I

Introduction to Python: Python variables, Python basic Operators, Understanding python blocks. Python Data Types, Declaring and using Numeric data types: int, float etc. Python Program Flow Control Conditional blocks: if, else and else if, Simple for loops in python, For loop using ranges, string, list and dictionaries. Use of while loops in python, Loop manipulation using pass, continue, break and else. Programming using Pythonconditional and loop blocks.

Unit II

Python Complex data types: Using string data type and string operations, Defining list and list slicing, Use of Tuple data type. String, List and Dictionary, Manipulations Building blocks of python programs, string manipulation methods, List manipulation. Dictionary manipulation, Programming using string, list and dictionary in-built functions. Python Functions, Organizing python codes using functions.

Unit III

Python File Operations: Reading files, Writing files in python, Understanding read functions, read(), readline(), readlines(). Understanding write functions, write() and writelines() Manipulating file pointer using seek Programming, using file operations

Unit IV

Python packages: Simple programs using the built-in functions of packages matplotlib, numpy, pandas etc. GUI Programming: Tkinter introduction, Tkinter and PythonProgramming, Tk Widgets, Tkinter examples. Python programming with IDE.

Prashant

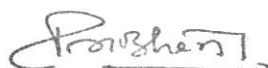
Course Outcomes

A student will be able to:

- Study fundamental concepts of Python.
- Understand the concept of packages and file handling in Python programming.
- Understand and apply concepts related to Graphical User Interface (GUI) programming in Python.

Suggested Readings:

1. Wesley J. Chun, "Core Python Applications Programming", 3rd Edition , Pearson Education, 2016
2. 2. Lambert, Fundamentals of Python: First Programs with MindTap, 2nd 1st edition , Cengage Learning publication
3. 3. Charles Dierbach, "Introduction to Computer Science using Python", Wiley, 2015
4. 4. Jeeva Jose &P.SojanLal, "Introduction to Computing and Problem Solving with PYTHON", Khanna Publishers, New Delhi, 2016
5. 5. Downey, A. et al., "How to think like a Computer Scientist: Learning with Python", John Wiley, 2015
6. 6. Mark Lutz, "Learning Python", 5th edition, Orelly Publication, 2013, ISBN 978-1449355739
7. 7. John Zelle, "Python Programming: An Introduction to Computer Science", Second edition, Course Technology Cengage
8. Learning Publications, 2013, ISBN 978- 1590282410



25BCA1ML- N-502	Computer Network	4L:0T:0P	4 Credits
-----------------	-------------------------	----------	-----------

Maximum Marks:100
External Examination: 70
Internal Assessment: 30
Max. Time: 3 Hrs

Course Objectives

- The Purpose of the course is to understand the evolution of computer networks using the layered network architecture

Note: There shall be nine questions in all. Question no. 1 shall be compulsory, consisting of eight short answer type questions covering the entire syllabus. Two questions will be asked from each unit. Students will have to attempt one question from each unit. Each question shall carry equal marks.

Unit I

Introduction to Computer Networks : Data Communication System and its components, Types of computer networks, Wireless and Wired networks, Network topologies, protocols, interfaces and services, ISO- OSI reference model, TCP/IP architecture.

Physical Layer: Concept of Analog & Digital Signal, Bandwidth, Transmission Multiplexing : Frequency Division, Time Division, Wavelength Division, FDMA, TDMA, CDMA, Transmission Media: Twisted pair, Coaxial cable, Fiber optics, Wireless transmission (radio, microwave, infrared), Switching: Circuit Switching, Message Switching ,Packet Switching.

Unit II

Data link layer: Error Control, Types of errors, framing(character and bit stuffing), error detection & correction methods; Flow control; Protocols: Stop & wait ARQ, sliding window protocols, HDLC;

Medium access sub layer: Point to point protocol, FDDI, token bus, token ring; Reservation, polling, Multiple access protocols: Pure ALOHA, Slotted ALOHA, CSMA, CSMA/CD,, Traditional Ethernet, fast Ethernet, Network devices-repeaters, hubs, switches, Bridges, Router, Gateway

Unit III

Network layer: Addressing : Internet address, sub-netting; Routing techniques, static vs. dynamic routing , routing table, DHCP,

Routing algorithms: shortest path algorithm, flooding, distance vector routing, link state routing; Protocols: ARP, RARP, IP, IPV6;

Unit IV

Transport & Application Layer: Process to process delivery; UDP; TCP, Congestion control algorithm: bucket algorithmS, Quality of service: techniques to improve QoS.

Application layer: DNS; SMTP, SNMP, FTP, HTTP & WWW; Firewalls, Bluetooth, Email, S/MIME IMAP

Network Security: Cryptography, user authentication, security protocols in internet, public key encryption algorithm, digital signatures

[Signature]

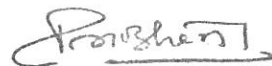
Course Outcomes

A student will be able to:

- Describe the ISO-OSI reference model and TCP/IP architecture, and demonstrate an understanding of the physical layer, including analog and digital signals, bandwidth, transmission impairments, multiplexing techniques, and different transmission media.
- Analyze the data link layer and medium access sub layer protocols and techniques for error control, flow control, framing, and multiple access, as well as knowledge of various network devices..
- Comprehend and implement network layer concepts including addressing, routing techniques, routing algorithms, protocols, and standards
- Understand the functions of the Transport layer, including process-to-process delivery, UDP, TCP, and congestion control algorithms
- Analyze and evaluate various security protocols used in the internet to ensure data confidentiality, integrity, and availability.

Suggested Readings:

1. Larry L.Peterson, Peter S. Davie, "Computer Networks", Elsevier, Fifth Edition, 2012.
2. William Stallings, "Data and Computer Communication", Eighth Edition, Pearson Education, 2007
3. Behrouz A. Forouzan, "Data communication and Networking", Tata McGraw Hill, Fourth Edition, 2011.
4. Computer Networks, 4th Edition, Pearson Education by Andrew S. Tanenbau



25BCAIML- N-503	Artificial Intelligence	4L:0T:0P	4 Credits
-----------------	-------------------------	----------	-----------

Maximum Marks:100
External Examination: 70
Internal Assessment: 30
Max. Time: 3 Hrs

Course Objectives

- The Purpose of the course is to introduce the principles and paradigms of programming languages for design and implement the software intensive systems.

Note: There shall be nine questions in all. Question no. 1 shall be compulsory, consisting of eight short answer type questions covering the entire syllabus. Two questions will be asked from each unit. Students will have to attempt one question from each unit. Each question shall carry equal marks.

Unit I

Introduction to artificial intelligence, Historical development and foundation areas of artificial intelligence,

Tasks and application areas of artificial intelligence.

Introduction, types and structure of intelligent agents, Computer Vision, Natural language processing.

Unit II

Introduction, Problem solving by searching, Searching for solutions, Uniformed searching techniques, Informed searching techniques, Local search algorithms, Adversarial search methods, Search techniques used in games, AlphaBeta pruning. Propositional logic, Predicate logic, First order logic, Inference in first order logic, Clause form conversion, Resolution.

Unit III

Chaining- concept, forward chaining and backward chaining, Utility theory and Probabilistic reasoning, Hidden Markov model, Bayesian networks. Introduction, types and application areas, Statistical learning methods- Regression Decision trees, Naïve Bayes models, Learning with hidden data- concept and EM algorithm, Reinforcement learning.

Unit IV

Introduction and design principles, Statistical pattern recognition, Parameter estimation methods - Principle component analysis and Linear discrimination analysis, Classification techniques - Nearest neighbour rule and Bayes classifier, K-means clustering, Support vector machine.

Prashant

Course Outcomes

A student will be able to:

- Write the meaning of intelligence and study various intelligent agents.
- Apply AI searching algorithms in different problem domains.
- Analyze various models for knowledge representations.
- Demonstrate the basic concepts of machine learning to analyze and implement widely used learning methods and algorithms.
- Outline the knowledge of pattern recognition.

Suggested Readings:

1. Khemani D., "A First Course in Artificial Intelligence", McGraw Hill.
2. Winston P. H., "Artificial Intelligence", Pearson Education.
3. Thornton C. and Boulay B., "Artificial Intelligence- Strategies, Applications and Models through Search", New Age International Publishers.
4. Russell S. and Norvig P., "Artificial Intelligence – A Modern Approach", Pearson Education.
5. Rich E. and Knight K., "Artificial Intelligence", McGraw Hill Publications.
6. Charnik E. and McDermott D., "Introduction to Artificial Intelligence", Pearson Education.

Prashant

25BCAIML- N- 504	Web Security	3L:0T:0P	3 Credits
------------------	--------------	----------	-----------

Maximum Marks:70
External Examination: 50
Internal Assessment: 20
Max. Time: 3 Hrs

Course Objectives

- Gain a comprehensive understanding of web security principles, including the importance of confidentiality, integrity, and availability.
- Explore secure authentication mechanisms and session management best practices to mitigate common vulnerabilities like session hijacking and fixation.
- Learn about common web vulnerabilities such as Cross-Site Scripting (CSS), Cross-Site Request Forgery (CSRF), and SQL Injection.
- Learn about security testing methodologies, tools, and techniques for vulnerability assessment and penetration testing.

Note: There shall be nine questions in all. Question no. 1 shall be compulsory, consisting of eight short answer type questions covering the entire syllabus. Two questions will be asked from each unit. Students will have to attempt one question from each unit. Each question shall carry equal marks.

Unit I

Introduction to Web Security, Importance of web security, Basic security concepts: confidentiality, integrity, availability. Overview of web application architecture, Common security terminology

Types of attackers and their motives, Common attack vectors, Case studies of recent web security breaches

Authentication and Session Management, Secure authentication mechanisms, Password policies and storage techniques, multi-factor authentication, Session management best practices, Common vulnerabilities (e.g., session hijacking, fixation)

Unit II

Cross-Site Scripting (XSS)-Types of XSS: reflected, stored, DOM-based, Detection and exploitation techniques, Preventive measures: input validation, output encoding

Cross-Site Request Forgery (CSRF)-Understanding CSRF attacks, Defense mechanisms: anti-CSRF tokens, Same Site cookies.

SQL Injection-Basics of SQL injection, Types of SQL injection: in-band, inferential, out-of-band, Detection techniques and tools, Mitigation strategies

Unit III

Secure Coding Practices- Principles of secure coding, Common coding flaws and how to avoid them using secure libraries and frameworks, Code reviews and static analysis tools
Security Testing and Vulnerability Assessment, Types of security testing: black-box, white-box, grey-box

Prashant

Tools for security testing (e.g., Burp Suite, OWASP ZAP), Conducting penetration tests, Reporting and remediating vulnerabilities.

Unit IV

HTTPS and Secure Communication- Importance of HTTPS, How SSL/TLS works, Certificate authorities and chain of trust, Implementing HTTPS in web applications, Common pitfalls and best practices, Content Security Policy (CSP) and Security Headers, Web Application Firewalls (WAF) and Security Tools, Machine learning and AI in web security.

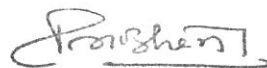
Course Outcomes

A student will be able to

- Develop a solid understanding of web security fundamentals, including confidentiality, integrity, and availability, and their importance in web application development.
- Understand best practices for password policies, storage techniques, and multi-factor authentication, ensuring robust security measures in authentication processes.
- Develop skills in identifying and remediating vulnerabilities through effective coding practices, testing methodologies, and security tools.
- Apply secure coding practices and conduct security testing using tools like Burp Suite and OWASP ZAP to identify and address vulnerabilities in web applications.

Suggested Readings:

1. "Web Application Security, A Beginner's Guide" by Bryan Sullivan and Vincent Liu.
2. "The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws" by Dafydd Stuttard and Marcus Pinto
3. "Computer Security: Principles and Practice" by William Stallings and Lawrie Brown.



Python Programming Lab (Based on 25BCAIML- N- 501)	0L:0T:2P	1 Credits
---	----------	-----------

Maximum Marks:30
External Examination: 20
Internal Assessment: 10

Lab 1: Basics of python

- Write a Python program to check if a given year is a leap year or not.
- Write a Python program to generate the Fibonacci series up to n terms.

Lab 2: Basics

- Declare a variable num_complex and assign it a complex number. Perform complex arithmetic operations like addition, subtraction, multiplication, and division with num_complex.
- Write a Python program to implement the Tower of Hanoi problem using recursion.
- Implement a Python program to generate Pascal's triangle up to a given number of rows.

Lab 3: String

- Write a Python function count_vowels that takes a string as input and returns the count of vowels (both uppercase and lowercase) present in the string.
- Write a Python function reverse_words that takes a sentence as input and returns the sentence with each word reversed.
- Write a Python function find_max_min that takes a list of numbers as input and returns a tuple containing the maximum and minimum numbers in the list.
- Write a Python function remove_duplicates that takes a list as input and returns a new list with duplicate elements removed while preserving the original order.

Lab 4: Dictionaries

- Write a Python function merge_dicts that takes two dictionaries as input and returns a new dictionary containing the merged key-value pairs from both dictionaries. If there are common keys, prioritize the values from the second dictionary.
- Write a Python function flatten_dict that takes a nested dictionary as input and returns a flattened dictionary where each key is a tuple representing the nested structure and the value is the corresponding value from the original dictionary.

Lab 5: Strings

Prashant

- Write a Python program that takes a sentence from the user, counts the frequency of each word, and prints the word-frequency pairs sorted by frequency in descending order.
- Write a Python program that takes a list of strings as input, converts each string to uppercase, removes any duplicates, and prints the modified list.

Lab 6: Files

- Write a Python program that reads the first 50 characters from a text file named "data.txt" and prints them to the console.
- Write a Python program that reads the contents of a text file named "numbers.txt" containing integers separated by spaces, converts them to a list of integers, and prints the sum of the numbers.
- Modify the program to read the contents of the file line by line and print the sum of each line.

Lab 7: Files

- Write a Python program that takes a list of strings as input and writes each string to a new line in a text file named "strings.txt".
- Modify the program to write the list of strings as a single line separated by commas in the "strings.txt" file.

Lab 8: Matplotlib

- Write a Python program using Matplotlib to plot a simple line graph showing the trend of temperature over a week. Use lists to represent days and corresponding temperatures.
- Extend the program to save the plot as a PNG file named "temperature_plot.png".

Lab 9: Pandas

- Write a Python program using Pandas to read a CSV file named "data.csv" containing information about students (e.g., name, age, grade) and display the first 5 rows of the DataFrame.
- Modify the program to calculate the average age of the students and print it.

Lab 10: Tkinter Examples

- Write a Python program using Tkinter to create a calculator application with buttons for digits, arithmetic operations, and a display for the result.
- Write another Python program using Tkinter to create a simple login form with entry widgets for username and password, and a button to submit the credentials.

Lab 11: Tkinter

- Write a Python program using Tkinter to create a simple window with a label widget displaying the text "Hello, Tkinter!".
- Add a button widget labeled "Click Me" to the window.

Prashant

Scheme of Examination for BCA (AI/ML)**Semester: 6th (w.e.f. Session 2025-26)**

S r . N o .	Course/ Paper Code	Title of the Course	Mode of Learnin g	Cours e Typ e	Credit			Contact Hours per Week			Examination Scheme			Total Mark s
					Theo ry	Practical / Seminar	Tot al	Theo ry	Practi cal /Semi nar	Tot al	Major Test (End Semest er Exa m)	Internal Assesse nt	Practical (External)	
1	25BCAIML-N- 601	Internship		CC	..	20	20	..	..	..	400	50		450
	Total						20							450

CC - Core Course

Note: Students will have to submit Internship Report after successful completion of Internship.



