MANGALORE UNIVERSITY

State Education Policy – 2024
[SEP-2024]

CURRICULUM STRUCTURE

FOR

BACHELOR OF COMPUTER APPLICATIONS
BCA-ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

MANGALORE UNIVERSITY

Suggested programme structure for the Under Graduate Programmes

[BCA, BCA (A.I and M.L), BCA (D.A)]

Semester	Course 1	Course 2	Course 3	Elective / Optional	Language	Compulsory	Total Credits	Total Working hour
I	5 (3T+2P)	5 (3T+2P)	5 T		3+3	2	23	4+4+4+4+5+4+4+2=31
II	5 (3T+2P)	5 (3T+2P)	5T		3+3	2	23	4+4+4+4+5+4+4+2=31
III	5 (3T+2P)	5 (3T+2P)	5T	2	3+3		23	4+4+4+4+5+4+4+2=31
IV	5 (3T+2P)	5 (3T+2P)	5T	2	3+3	2	25	4+4+4+4+5+2+4+4+2=33
V	8[(2x3T)+2P]	8[(2x3T)+2P]	8[(2x3T)+2P]			2	26	3+3+4+3+3+4+3+3+4+2=32
VI	3T	3T	3T	3T	Research Methodology + Project work 12		24	3+3+3+3+24=36
Total Credits for the Programme							**144**	

Note:

- Course1 and Course2: I to IV Semester: Theory 3 credits = 4 contact hours and Practical 2 credits=4 contact hours
- Course3: I to IV Semester: Theory 5 credits = 5 contact hours
- Course1, Course2 and Course3: V and VI Semester: Theory 3 credits = 3 contact hours and Practical 2 credits = 4 contact hours
- Elective/Optional: 2 credits=2 contact hours
- Languages: 3 credits=4 contact hours
- Compulsory: 2 credits=2 contact hours

CURRICULUM STRUCTURE FOR III AND IV SEMETER
BCA- ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Semester III								
Sl. No	Course Code	Title of the Course	Category of Courses	Teaching Hours per Week	SEE	IA	Total Marks	Credits
1		LANGUAGE-I	Lang	4	80	20	100	3
2		LANGUAGE-II	Lang	4	80	20	100	3
3	BCAAIMS301	Object Oriented Programming using Java	Core	4	80	20	100	3
4	BCAAIMS302	Operating System	Core	4	80	20	100	3
5	BCAAIMS303	Computer Network	Core	5	80	20	100	5
6	BCAAIPS304	Object Oriented Programming Lab	Practical	4	40	10	50	2
7	BCAAIPS305	Operating System Lab	Practical	4	40	10	50	2
8	BCAAIES301 BCAAIES302 BCAAIES303	A) Digital Marketing B) Web Content Management System C) DEVOPS	Elective	2	40	10	50	2
Sub - Total				31	520	130	650	23

Semester IV								
Sl. No	Course Code	Title of the Course	Category of Courses	Teaching Hours per Week	SEE	IA	Total Marks	Credits
1		LANGUAGE-I	Lang	4	80	20	100	3
2		LANGUAGE-II	Lang	4	80	20	100	3
3	BCAAIMS401	Artificial Intelligence-I	Core	4	80	20	100	3
4	BCAAIMS402	Python Programming	Core	4	80	20	100	3
5	BCAAIMS403	Software Engineering	Core	5	80	20	100	5
6	BCAAIPS404	Artificial Intelligence-I Lab	Practical	4	40	10	50	2
7	BCAAIPS405	Python Programming Lab	Practical	4	40	10	50	2
8	BCAAIES401 BCAAIES402 BCAAIES403	A) Cloud Computing B) Object Oriented Analysis and Design C) Digital Image Processing	Elective	2	40	10	50	2
9	BCAAISS401	Internet Basics	Compulsory	2	40	10	50	2
			Sub - Total	33	640	160	800	25

Pedagogy: Lecture / PPT / Videos / Demonstration / Concept mapping / Case Studies examples / Tutorial / Problem Solving / Trouble shooting

SEMESTER III

Program Name	BCA-AIML	Semester	III
Course Title	**Object Oriented Programming using JAVA (THEORY)**		
Course Code:	**BCAAIMS301**	No. of Credits	03
Contact hours	52 Hours	Duration of SEE/Exam	3 Hours
Formative Assessment Marks	20	Summative Assessment Marks	80

Course Outcomes (COs):

After completing this course satisfactorily, a student will be able to:

- Understand the features of JAVA and the architecture of JVM
- Write, compile, and execute JAVA programs that may include basic data types and control flow constructs and how type casting is done Identify classes, objects, members of a class and relationships among them needed for a specific problem and demonstrate the concepts of polymorphism and inheritance
- The students will be able to demonstrate programs based on interfaces and threads and explain the benefits of JAVA's Exceptional handling mechanism compared to other Programming Language
- Write, compile, execute JAVA programs that include GUIs

Unit	Description	Hours
1	**Fundamentals of Object-Oriented Programming:** Introduction, Object Oriented Paradigm, Basic Concepts of OOP, Benefits and Applications of OOP. **Introduction to JAVA:** JAVA Features, JAVA Environment, Simple JAVA Program, JAVA Program Structure, JAVA Tokens, JAVA Statements, JAVA Virtual Machine. **JAVA Programming Basics:** Constants, Variables, Data Types, Declaration of variables, Giving values to the variable, Scope of variables, Symbolic constants, Type casting. **Operators and Expressions:** Arithmetic Operators, Relational Operators, Logical Operators, Assignment Operator, Increment and Decrement Operators, Conditional Operator, Special Operators. **Using I/O:** Byte streams and character streams, predefined streams, reading console input, reading characters, strings, writing console output. **Decision Making and Branching:** Simple if statement, if..else statement, nesting of if..else, the else..if ladder, the Switch statement.	**13**
2	**Decision making and Looping** -The while statement, the do statement, the for statement . Jumps in loops, Labelled loops. **Class and Objects** - Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, the 'this' keyword, Overloading Methods, Using Objects as Parameters, Returning Objects, Recursion, Understanding 'static', Introducing 'final ', Using Command-Line Arguments, Varargs : Variable-Length Arguments **Arrays and Strings:** One dimensional arrays, Creating an array, Two dimensional arrays, Strings	**13**

3	**Inheritance** - Inheritance Basics, using 'super', Creating Multilevel hierarchy, Method Overriding, Using Abstract Classes, Using final with Inheritance. **Packages and Interfaces** - Packages, Access protection in packages, Importing Packages, Interfaces. **Exception Handling** - Exception Handling Fundamentals – Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch clauses, finally, JAVA's built-in Exceptions **Multithreaded Programming-** Introduction, Creating threads, Extending the thread class, stopping and blocking thread, Life cycle of a thread.	**13**
4	**Event and GUI programming: Event Handling** - The delegation event model, Event Classes –ActionEvent, KeyEvent and MouseEvent Classes, Event Listener Interfaces –ActionListener, KeyListener and MouseListener interfaces. **Introducing Swing** – two key swing features, components and containers, the swing packages, a simple swing application **Exploring Swing-** Jlabel, JTextField, JButton, Checkboxes , Radio buttons , Jlist , JComboBox.	**13**

Textbooks :

1. E Balagurusamy, Programming with JAVA – A Primer, Fourth Edition, Tata McGraw Hill Education Private Limited.
2. Herbert Schildt, JAVA: The Complete Reference, 7th Edition, MGH Publication.

Reference Books:

1. Herbert Schildt, JAVA 2-The Complete Reference, Fifth Edition, McGraw Hill publication.
2. Cay S. Horstmann, Core JAVA Volume I–Fundamentals, Prentice Hall.
3. Somashekara, M.T., Guru, D.S., Manjunatha, K.S, Object Oriented Programming with JAVA, EEE Edition, PHI.

Program Name	BCA-AIML	Semester	III
Course Title	**Operating System (THEORY)**		
Course Code:	**BCAAIMS302**	No. of Credits	03
Contact hours	52 Hours	Duration of SEE/Exam	3 Hours
Formative Assessment Marks	20	Summative Assessment Marks	80

Course Outcomes (COs):

At the end of the course, students will be able to:

- Understand the fundamentals of the operating system.
- Comprehend multithreaded programming, process management, process synchronization, memory management and storage management.
- Compare the performance of Scheduling Algorithms
- Identify the features of I/O and File handling methods.

Unit	Description	Hours
1	**Introduction:** Operating System, Simple Batch Systems, Multi programmed Batched Systems, Time Sharing Systems, Real-Time Systems, Multi-processor Systems. **System Components, Operating System Services. Process:** Process Concept, Process Scheduling, Cooperating Process, **Threads** (Thread Concept, Single and Multiple Threads, Benefits): **CPU Scheduling:** Basic Concepts, Scheduling Criteria, Scheduling Algorithms.	**13**
2	**Process Synchronization**: Introduction Race Condition, Critical Section, Semaphores; Synchronization in Linux. **Deadlocks:** Deadlock Characterization, Methods of Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock	**13**
3	**Memory Management.** Logical versus Physical Address Space, Swapping, Contiguous Allocation (Memory Allocation, Fragmentation), Paging (Basic Method), Segmentation (Basic Method). **Virtual Memory.** Demand Paging, Page Replacement, Page Replacement Algorithms, Thrashing (concept). **File System.** File Concept, Access Methods **Disk Scheduling (I/O Management):** Introduction and Scheduling Algorithm	**13**

4	**Linux files system**, login and logout. **Linux commands:** Directory oriented command, wild card characters, File oriented commands, File Access Permissions, Process oriented commands, Background processing, Communication oriented commands, General purpose commands, **Pipe and Filter** related commands, **vi editor, Shell programming**,	**13**

Text Books:

1. Abraham Silberschartz and Peter Galvin, **Operating System Concepts** ,6th edition,TMH
2. K.L. James, **Linux: Learning the Essentials**, PHI learning private limited,2011
3. B Mohammed Ibrahim, **Linux: A Practical Approach**, FireWallMedia,2009

Reference Books:

1. Andrew S Tanenbaum, **Operating System Design and Implementation**, PHI
2. Milan Milenkovic, **Operating Systems**, TMH
3. Cristopher Negus, Dreamtech, **RedHatLinux9 Bible**, Wiley Publication
4. Shital Vivek Ghate , Operating System Concepts and Basic Linux Commands ; Educreation Publishing.
5. Richard Fox , LINUX with Operating System Concepts, CRC PressMark
6. G Sobell, "**A Practical Guide to Linux**", PEARSON Publishers

Program Name	BCA-AIML	Semester	III
Course Title	**Computer Networks (THEORY)**		
Course Code:	**BCAAIMN303**	No. of Credits	05
Contact hours	60 Hours	Duration of SEE/Exam	3 Hours
Formative Assessment Marks	20	Summative Assessment Marks	80

Course Outcomes (COs):

At the end of the course, students will be able to:

- Explain the transmission technique of digital data between two or more computers and a computer network that allows computers to exchange data.
- Apply the basics of data communication and various types of computer networks in real world applications.
- Compare the different layers of protocols.
- Compare the key networking protocols and their hierarchical relationship in the conceptual model like TCP/IP and OSI.

Unit	Description	Hours
1	**Introduction:** Uses of Computer Networks-Business Applications, Home Applications, Mobile Users, Social Issues; Network Hardware-Local Area Networks, Metropolitan Area Networks, Wide Area Networks, Wireless Networks, Home Networks, Internetworks. Network Software- Reference Models-the OSI Reference Model, The TCPIP Reference Model, A Comparison of the OSI and TCP/IP Reference Models.	**15**
2	**The Physical Layer**-Transmission Media-Twisted-Pair, Coaxial Cable, Fiber Optics, Multiplexing (FDM, TDM) Switching **Data Link Layer:** Design Issues-Services Provided to the Network Layer, Framing, Error Control, Flow Control, Error Detection and Correction, Error correcting codes, Error detecting Codes.	**15**
3	**The Network Layer:** Network Layer design issues, Store and Forward Packet switching, Services Provided to the Transport Layer, Implementation of Connectionless Service, Implementation of Connection-Oriented Service, Comparison of Virtual Circuit and Datagram. Internetworking-How Networks can be connected, Connectionless Internetworking, Tunnelling, Internetwork Routing, Fragmentation. The Network Layer in the Internet-the IP Protocol, IP Addresses, OSPF, BGP, Internet Multicasting, IPv6.	**15**
4	**The Transport Layer**: The Transport Service-Services Provided to the Upper Layers, Transport Service Primitives, Elements of Transport Protocols-Addressing, Connection Establishment, Connection Release. The Internet Transport Protocols-UDP-Introduction to UDP, Remote Procedure Call, The Real-Time Transport Protocol; TCP-Introduction to TCP, The TCP Service Model, The TCP Protocol, The TCP Segment Header, TCP Connection Establishment, TCP Connection Release.	**15**

	The Application Layer-DNS-The DNS Name Space, Resource Records, Name Servers. Electronic Mail-Architecture and Services, The User Agent, Message Formats, Message Transfer, Final Delivery;	

Text Book:

1. Computer Networks, Andrew S. Tanenbaum, 5th Edition, Pearson Education, 2010.

Reference Books:

1. Data Communication and Networking, Behrouza A Forouzan, 3rd Edition, Tata McGraw Hill, 2001.
2. Data and Computer Communications, William Stallings, 10th Edition, Pearson Education, 2017.
3. Data Communication and Computer Networks, Brijendra Singh, 3rd Edition, PHI, 2012.
4. Data Communication and Network, Dr. Prasad, Wiley Dreamtech.

Program Name	**BCA-AIML**	Semester	**III**
Course Title	**Object Oriented Programming Lab using JAVA**		
Course Code:	**BCAAIMPS304**	No. of Credits	**02**
Contact hours	**52 Hours**	Duration of SEE/Exam	**3 Hours**
Formative Assessment Marks	**10**	Summative Assessment Marks	**40**

PART-A

1. Write a Program to Program to accept student name and marks in three subjects. Find the total marks, average and grade (depending on the average marks).
2. Write a Program to Program to input Start and End limits and print all Fibonacci numbers between the ranges.(Use for loop)
3. Design a class: ShowRoom with the following description:

   Instance variables / Data members:
   String name — To store the name of the custome
   long mobno — To store the mobile number of the customer
   double cost — To store the cost of the items purchased
   double dis — To store the discount amount
   double amount — To store the amount to be paid after discount
   Member methods:
   ShowRoom() — default constructor to initialize data members
   void input() — To input customer name, mobile number, cost
   void calculate() — To calculate discount on the cost of purchased items, based on following criteria
   void display() — To display customer name, mobile number, amount to be paid after discount.

Cost	Discount
Less than or equal to ₹10000	5%
More than ₹10000 and less than or equal to ₹20000	10%
More than ₹20000 and less than or equal to ₹35000	15%
More than ₹35000	20%

Write a main method to create an object of the class and call the above member methods. [Class Basics]

4. Write a Program to create a class DISTANCE with the data members feet and inches. Use a constructor to read the data and a member function Sum ( ) to add two distances by using objects as method arguments and show the result. (Input and output of inches should be less than 12).
5. Write a Program to create a class “Matrix” that would contain integer values having varied numbers of columns for each row. Print row-wise sum.

6. Define a class to accept a String and print the number of digits, alphabets and special characters in the string.

   Example: S = "Kapil Dev@83"
   Output:

   Total number of characters - 12
   Number of digits – 2
   Number of upper-case alphabets – 2

   Number of lower-case alphabets - 6
   Number of Special characters – 1

7. Write a Program to add, remove and display elements of a Vector.

PART-B

1. Create a class named 'Member' having data members: Name, Age, PhoneNumber, Place and Salary. It also has a method named 'printSalary' which prints the salary of the members. Two classes 'Employee' and 'Manager' inherit the 'Member' class. The 'Employee' and 'Manager' classes have data members 'specialization' and 'department' respectively. Now, assign name, age, phone number, address and salary to an employee and a manager by making an object of both of these classes and print the same. [inheritance]

2. Write a Program to implement the following class hierarchy: Student: id, name StudentExam (derived from Student): Marks of 3subjects, total marks StudentResult (derived from StudentExam) : percentage, grade Define appropriate methods to accept and calculate grade based on existing criteria and display details of N students

3. Write a Program to calculate marks of a student using multiple inheritance implemented through interface

   Create a class called student with
   a) Data members namely, name and regno for keeping Name and RegNo of a Student
   b) Member functions namely, getdata() and putdata() for initializing and displaying the contents of data members
   Create a class called mark which inherits the properties of student class with
   a) Data members namely, m1, m2 and m3 for keeping marks of three subjects
   b) Member functions namely, getmarks( ) and putmarks( ) for initializing and displaying the contents of data members
   Create an Interface Student, with a final data member pract_mark for keeping practical mark and initialize it and member method putPractMarks() to display it
   Create a class called Result which inherits the properties of mark class and implements the interface student, with Data member total and Member function display( ) for computing total marks and checking whether a student is passed or failed in based on each subject. The display() function computes the total marks as total=m1+m2+m3+pract_mark. It checks for the condition ((m1 > 40) && (m2 > 40) && (m3 > 40) && (pract_mark > 20)). If the condition is True, then print that the result as PASS else print that the result as FAIL.

4. Write a Program to create an abstract class named shape that contains two integers, and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Ellipse such that

each one of the classes extends the class shape. Each one of the class contains only the method printArea() that print the area of the given shape.[Abstract class]

5. Create a package named FINANCE to encapsulate functionality for calculating compound interest and simple interest. Within the FINANCE package, define a class containing a method to calculate compound interest. Also, within the FINANCE package, define another class containing a method to calculate simple interest.
   Create a package EKYC to encapsulate for account holders profile creation. Within EKYC package define class PERSONAL_INFO containing members(Name, Adhar number ,PAN, Mobile number and Address) and methods(input () and print()). Also within EKYC package define a class ACCOUNT_INFO containing members(ACCOUNT_NUM, CUSTOMER_ID, BALANCE) and methods(input () and print()).

   Create a main class BANK that resides outside the FINANCE AND EKYC package. Inside the main method, invoke the respective methods from the packages to perform the interest calculations and display all the details of bank customer.

6. Write a Program that creates a user interface to perform basic integer operations such as addition, subtraction, multiplication and division. The user enters two numbers in the TextFields - Num1 and Num2. The result of operations must be displayed in the ResultTextField when the operation(+ or – or x or /) button is clicked. Appropriate Exception handling message to be displayed in the Result TextField when Num1 or Num2 is not an integer or Num2 is Zero when division operation is applied.

7. Using the swing components, design the frame for shopping a book that accepts book code, book name, and Price. Calculate the discount on code as follows.

Code	Discount rate
101	15%
102	20%
103	25%
Any other	5%

   Find the discount amount and Net bill amount. Display the bill.

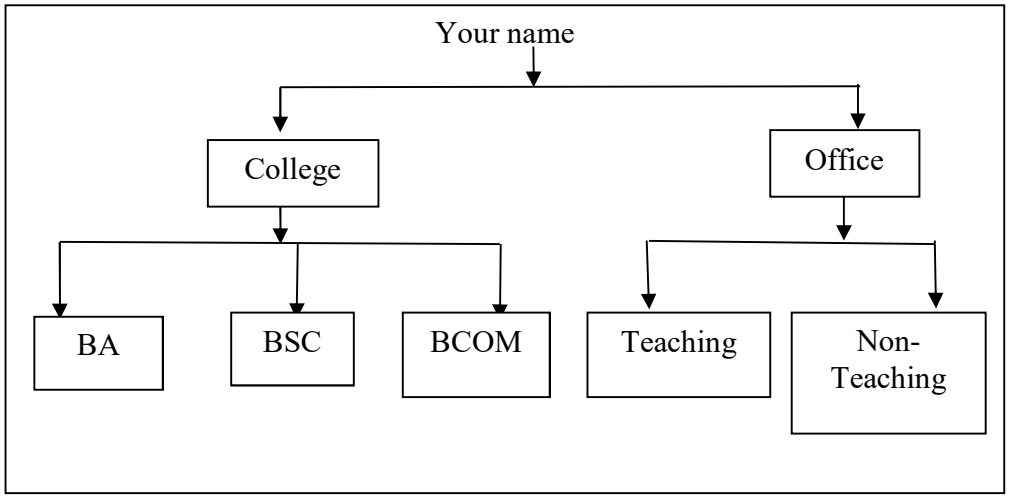
Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	PART-A:Writing:5 Marks Execution:3Marks	15 Marks
Program-2	PART-B:Writing:8 Marks Execution:4Marks	20 Marks
Practical Record		05 Marks
Total		40Marks

Program Name	**BCA-AIML**	Semester	**III**
Course Title	**Operating System Lab**		
Course Code:	**BCAAIMP305**	No. of Credits	02
Contact hours	52 Hours	Duration of SEE/Exam	3 Hours
Formative Assessment Marks	10	Summative Assessment Marks	40

PART-A

1. Use Linux commands to create the directory structure and perform the listed actions.



   Put one file each inside the directories BA, BCom, Teaching and Non-Teaching with meaningful content.
   a) Goto BA directory and then move its file to non-teaching directory.
   b) Change the directory to non-teaching from root directory using only single command.
   c) From BA copy the content to B.Com. Delete BA directory recursively
2. Consider the above directory structure. Write and execute Linux commands to
   a) List the files in long format. Starting from root directory including sub directory contents
   b) Count how many files exist in office directory.
   c) Display the present path.
   d) Display current date in the form MM/DD/YY HH:MM:SS
3. Write a Linux shell script to create a file which contains students' information such as Rollno, Name, Gender, class (BCA, BSc, BCom) and Total marks.
   a) Append the 2 more records.
   b) Display the contents.
   c) Display the records of only BSc course and sort on reverse order of name and store in BSCnew.txt.
   d) Display the contents of BSCnew.txt

e) Rename the student.dat to stud.dat

4. Using the above stud.dat file, Write Linux shell scripts to
   a) Add the following permission
      i. Execute for user.
      ii. Remove write permission for group
      iii. Check the changes in the permission by listing the file.
   b) Select only the female names and store in female.dat and male names to male.dat
   c) Combine these names from both the files .Show the new file.
   d) Change the case of alphabets of the contents in male.dat.
   e) List all the files that start with "B" in the current directory.
5. Write a Linux shell script to Accept 'n' and check whether the number is a prime or not.
6. Write a Linux shell script to Accept 'n' and find the sum of the series 1!+3!+5!.....+n!.
7. Write a Linux shell script to Display all natural numbers between two integers and also find their sum.

PART B

1. Write a shell script to accept 'n' integers and count the positive, negative integers and zeros separately. Also find the sum of the positive and negative integers separately.
2. Write a shell script to accept many characters and count individual vowels, digits, spaces, special characters and consonants.
3. Write a Linux shell script to accept student name and marks in 3 subjects through command line arguments. Find Total marks, Average marks and Result(PASS if marks in each subject is more than or equal to 40 and FAIL if mark in any subject is less than 40) Grade (depending on average marks).
4. Write a Linux shell script to Accept a word and check whether it begins with lowercase vowel or uppercase vowel, ends with a digit or whether it is a three-letter word.
5. Write a menu driven shell script for the following.
   a) Display the current working directory.
   b) Rename a file(check for the existence of the source file)
   c) List the users logged in.
   d) Append the contents of a file to another file(display the message if the file doesn't exist in the directory).
6. Write a menu driven shell script for the following.
   a) Rename a file (check for the existence of the source file)
   b) List all file names/ directory names in the present working directory which has the specified pattern
   c) List of directory having all the permission
   d) List only files names in long format.
7. Write a shell script to accept many filenames through command line. Do the following for each filename
   a) If it is an ordinary file, display its content and also check whether it has Execute permission.
   b) If it is directory, display the number of files in it.
   c) If the file/directory does not exist, display a message

Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	PART-A Writing: 5 Marks Execution: 3 Marks	15 Marks
Program-2	PART-B Writing: 8 Marks Execution: 4 Marks	20 Marks
Practical Record		05 Marks
Total		40 Marks

Program Name	BCA-AIML	Semester	**III**
Course Title	**Digital Marketing (ELECTIVE)**		
Course Code:	**BCAAIMES301**	No. of Credits	02
Contact hours	26 Hours	Duration of SEE/Exam	2 Hours
Formative Assessment Marks	10	Summative Assessment Marks	40

Course Outcomes (COs):

After the successful completion of the course, the student will be able to:

- Understand the fundamental concepts and principles of digital marketing.
- Develop practical skills to implement various digital marketing strategies and techniques
- Analyze and evaluate the effectiveness of digital marketing campaigns.
- Create comprehensive digital marketing plans and strategies.

Unit	Description	Hours
1	**Introduction to Digital Marketing:** Overview of digital marketing, Evolution of digital marketing, Importance and benefits of digital marketing, Digital marketing channels and platforms **Digital Marketing Strategy and Planning:** Developing a digital marketing strategy, Setting goals and objectives, Budgeting and resource allocation. Campaign planning and execution, Monitoring and adjusting digital marketing campaigns	**8**
2	**Social Media Marketing:** Overview of social media marketing, social media platforms and their features, Creating andoptimizing social media profiles, social media content strategy, social media advertising and analytics **Email Marketing:** Introduction to email marketing, building an email list, Creating effective email campaigns, Email automation and segmentation, Email marketing metrics and analytics.	**8**
3	**Mobile Marketing:** Mobile marketing overview, Mobile advertising strategies, Mobile app marketing, Location-based marketing, Mobile marketing analytics **Analytics and Reporting:** Importance of analytics in digital marketing, setting up web analytics tools (e.g., Google Analytics), Tracking and measuring key performance indicators (KPIs), Conversion tracking and optimization, Reporting and data visualization	**10**

Text Book:

1. "Digital Marketing Strategy: An Integrated Approach to Online Marketing" by Simon Kingsnorth.

Reference Books:

1. "Email Marketing Rules: How to Wear a White Hat, Shoot Straight, and Win Hearts" by Chad S. White

2. "Content Inc.: How Entrepreneurs Use Content to Build Massive Audiences and Create Radically Successful Businesses" by Joe Pulizzi
3. "Mobile Marketing: How Mobile Technology is Revolutionizing Marketing, Communications and Advertising" by Daniel Rowles
4. "Web Analytics 2.0: The Art of Online Accountability and Science of Customer Centricity" by Avinash Kaushik

Program Name	BCA-AIML	Semester	**III**
Course Title	**Web Content Management System(ELECTIVE)**		
Course Code:	**BCAAIMES302**	No. of Credits	02
Contact hours	26 Hours	Duration of SEE/Exam	2 Hours
Formative Assessment Marks	10	Summative Assessment Marks	40

Course Outcomes (COs):

After the successful completion of the course, the student will be able to:

- Understand content development basics.
- Gain Knowledge of tools for multimedia content development for audio/ video, graphics, animations, presentations, screen casting.
- Host websites and develop content for social media platforms such as wiki and blog
- Understand e-publications and virtual reality .
- Understand the use of e-learning platform Moodle and CMS applications Drupal and Joomla

Unit	Description	Hours
1	Web Content Development and Management, Content Types and Formats, Norms and Guidelines of Content Development, Creating Digital Graphics, Audio Production and Editing.	**8**
2	Web Hosting and Managing Multimedia Content, Creating and Maintaining a Wiki Site. Presentation Software Part , Screen casting Tools and Techniques, Multilingual Content Development.	**8**
3	Planning and Developing Dynamic Web Content Sites, Website Design Using CSS Creating and Maintaining a WIKI Site, Creating and Managing a Blog Site. Content Management System: Joomla Content Management System: Drupal	**10**

Text Books:

1. Web Content Management: Systems, Features, and Best Practices 1st Edition by Deane Barker.
2. Content Management Bible (2nd Edition) 2nd Edition by Bob Boiko.
3. Using Joomla!: Efficiently Build and Manage Custom Websites 2nd Edition by Ron Severdia

Program Name	**BCA-AIML**	Semester	**III**
Course Title	**DEVOPS (ELECTIVE)**		
Course Code:	**BCAAIMES303**	No. of Credits	**02**
Contact hours	**26 Hours**	Duration of SEE/Exam	**2 Hours**
Formative Assessment Marks	**10**	Summative Assessment Marks	**40**

Course Outcomes (COs):
After the successful completion of the course, the student will be able to:

- Understand the scalable VDI environment, addressing challenges such as boot storms and hardware limitations.
- Understand how to apply various DevOps tools to streamline and automate the software development lifecycle, including infrastructure as code and deployment automation.
- Understand how to utilize cloud services (IaaS, PaaS, Hybrid Cloud) to enhance DevOps practices, enabling full-stack deployments and efficient resource management.
- Understand how to integrate DevOps with ALM processes to improve the development, deployment, and management of mobile and multi-tier applications, scaling Agile methodologies across the enterprise.

Unit	Description	Hours
1	**Introduction to DevOps:** Business needs for DevOps, Business values for Devops, How DevOps works. **DevOps Capabilities:** Paths to DevOps Adoption, Plan, Devlop/Test, Deploy, Operate **Adopting DevOps:** Where to Begin, People in DevOps, Process in DevOps, Technology in DevOps	**8**
2	**Using Cloud in DevOps:** Cloud as DevOps enabler, Full Stack Deployments, cloud service model for DevOps, Hybrid Cloud **Using DevOps to solve Challenges:** Mobile applications, ALM processes, Scaling Agile, Multiple Tier Applications, DevOPs in the enterprise, Supply Chains, IOT. **DevOps Case Study:** Executive's Role, putting together a team, setting DevOps Goals, learning from the DevOps transformation, looking at the DevOps results. **DevOps Myths.**	**8**
3	**Basics of DevOps tools:** Git, Jenkins, Git lab, Docker, Kubernetes, Ansible, Terraform, Grafana, Sonar Qube	**10**

Textbooks:

1. Real-World DevOps Practices Paperback – 17 October 2024 by B. Thangaraju – Wiley Publishers, 2024
2. "DevOps For Dummies" by Sanjeev Sharma and Bernie Coyne. 2nd IBM Limited edition.

Reference Books:

1. Learning DevOps - Second Edition: A comprehensive guide to accelerating DevOps culture adoption with Terraform, Azure DevOps, Kubernetes, and Jenkins By: Mikael Krief (Author) | Publisher: Packt Publishing Limited
2. DevOps For Beginners: A Step-By-Step Guide To DevOps Best Practices By: Liam Foster
3. Learning DevOps: Jenkins, Kubernetes, Terraform, Azure DevOps. Jenkins, M. K. Packt Publishing Limited.
4. Joakim Verona (2016).Practical DevOps. Packt Publishing Limited
5. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology" by Gene Kim, Jez Humble, Patrick Debois, John Willis
6. The Phoenix Project: A Novel about IT, DevOps, and Helping Your Business Win" by Kim, Behr, Spafford

SEMESTER IV

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Artificial Intelligence – I(THEORY)**		
Course Code:	**BCAAIMS401**	No. of Credits	**03**
Contact hours	**52Hours**	Duration of SEE/Exam	**3 Hours**
Formative Assessment Marks	**20**	Summative Assessment Marks	**80**

Course Outcomes (COs):

At the end of the course, students will be able to:

- Identify and describe different types of intelligent agents and their environments, and implement basic agent-based systems.
- Formulate problems and apply both uninformed and informed search strategies to solve them.
- Represent knowledge using propositional and first-order logic and perform inference using these representations.
- Understand and implement fundamental machine learning algorithms such as linear regression, decision trees, and neural networks.
- Demonstrate knowledge of advanced AI topics, including reinforcement learning, natural language processing, and robotics.
- Analyze and discuss the ethical considerations and societal impacts of AI technologies.

Unit	Description	Hours
1	**Introduction-** What is AI, Foundations of Artificial Intelligence, History of Artificial Intelligence, State of the Art. **Intelligent Agents:** Agents and Environments, Good Behavior: The Concept of Rationality, Nature of Environments, Structure of Agents.	**13**
2	**Solving Problems by Searching**: Problem-Solving Agents, Example Problems, Searching for Solutions, Uninformed Search Strategies, Avoiding Repeated States, Searching with Partial Information **Informed Search and Exploration:** Informed (Heuristic) Search Strategies, Heuristic Functions, Local Search Algorithms and Optimization Problems	**13**
3	**Logical Agents:** Knowledge –Based Agents, The Wumps World, Logic, Propositional Logic: A Very Simple Logic, Reasoning Patterns in Propositional Logic, Effective propositional inference, Agents Based on Propositional Logic. **First Order Logic:** Representation Revisited, Syntax and Semantics of First Order Logic, Knowledge Engineering in First-Order Logic. **Inference in First-Order Logic:** Propositional vs. First-Order Inference, Unification and Lifting, Forward Chaining, Backward Chaining.	**13**

4	**Planning** –The Planning Problem, Planning with State-Space Search, Partial-Order Planning, Planning Graphs, Planning with Proposition Logic, Analysis of Planning Approaches. **Applications of AI** - Natural Language Processing, Text Classification and Information Retrieval, Speech Recognition , Image processing and computer vision, Robotics	**13**

Text Book:

1. Stuart Russel, Peter Norvig: Artificial Intelligence A Modern Approach, 2nd Edition, Pearson Education, 2003

Reference Books:

1. Tom Mitchell, "Machine Learning", 1st Edition, McGraw-Hill,2017
2. Elaine Rich, Kevin Knight, Shivashankar B Nair: Artificial Intelligence, Tata McGraw Hill 3rd edition,

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Python Programming (THEORY)**		
Course Code:	**BCAAIMS402**	No. of Credits	**03**
Contact hours	**52Hours**	Duration of SEE/Exam	**3 Hours**
Formative Assessment Marks	**20**	Summative Assessment Marks	**80**

Course Outcomes (COs):

At the end of the course, students will be able to:

- Explain the basic concepts of Python Programming.
- Demonstrate proficiency in the handling of loops and creation of functions.
- Identify the methods to create and manipulate lists, tuples and dictionaries.
- Discover the commonly used operations involving file handling.
- Interpret the concepts of Object-Oriented Programming as used in Python.
- Develop the emerging applications of relevant fields using Python.

Unit	Description	Hours
1	**Introduction to Python-** Features of Python, Flavors of python, Python Virtual machine, Memory management, Garbage Collection, Comparison between Python and C, JAVA and Python, Installing Python for Windows, Writing and executing Python program. **Python Basics**: Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Association, Data Types, Indentation, Comments, Console Input and Console Output, Type Conversions. **Input and Output**: Input/output Statements, Command line arguments. **Control Statements** – if, if..else, if..elif, while loop, for loop, else suite, break, continue, assert, return Statements **Arrays in Python**- Creating arrays, importing array module, Indexing and slicing on arrays, types of arrays, working with arrays using numpy.	**13**
2	**Functions** – Functions and methods, Defining, calling functions, returning multiple values, formal and actual parameters, Keyword argument Default arguments and variable argument, Local and Global variables, Anonymous functions and Lambdas, Decorators, and Generators, matrices in numpy, matrix functions, random numbers **Strings**: Creating and Storing Strings, Accessing Sting Characters; the str()function; Operations on Strings-Concatenation, Comparison, Slicing and Joining, Traversing, Python String Methods **Lists**: Creating Lists, Operations on Lists, Built-in Functions on Lists, Implementation of Stacks and Queues using Lists, Nested Lists	**13**

	Dictionaries: Creating Dictionaries, Operations on Dictionaries, Built-in Functions on Dictionaries, Dictionary Methods, Populating and Traversing Dictionaries. **Tuples and Sets**: Creating Tuples, Operations on Tuples, Built-in Functions on Tuples, Tuple Methods, Creating Sets, Operations on Sets, Built-in Functions on Sets, Set Methods.	
3	**File Handling**: File Types, Operations on Files– Create, Open, Read, Write, Close Files, File Names and Paths **Classes and Objects**-Defining classes and Objects, constructors, types of methods and variables, Inner classes. **Inheritance and Polymorphism**: Type of Inheritance, super () method, method overloading and Overriding, Operator Overloading **Exception Handling** –Type of exceptions, assert Statement, Except Block, User defined exceptions, logging the exceptions. **Regular expressions**: Sequence characters, Quantifiers and Special characters in regular expressions.	**13**
4	**Graphical User Interface**: Root window, font and colors, Canvas and frames. Widgets: Button, Label, Message, Text, Scrollbar, Checkbutton, Radiobutton, Entry, Spinbox, Listbox and Menu, Creating Tables. **Database Connectivity**: Types of databases used with Python, Using MySQL from Python, Retrieving and Inserting, updating and deleting data in a table, Creating Database tables through Python. Using Oracle database from Python Stored Procedures **Data Science Using Python**: Introduction, Data Frame, Data Visualization, Machine Learning	**13**

Text Books:

1. Dr. R. Nageswara Rao, Core Python Programming (2nd ed.). DreamTech Press.
2. Gowrishankar S., and Veena A. (2019). Introduction to Python Programming. CRC Press.
3. Dr. R. Nageswara Rao,(2021). Core Python Programming (3rd ed.). DreamTech Press.

Reference Books:

1. Brown, M. C. (2018). Python: The complete reference. McGraw Hill Education
2. Summerfield, M. (2010). Programming in Python 3: A complete introduction to the Python language (2nd ed.). Addison-Wesley.
3. Zelle, J. M. (2010). Python programming: An introduction to computer science (2nd ed.). Franklin, Beedle and Associates Inc.
4. Lutz, M. (2013). Learning Python (5th ed.). O'Reilly Media.
5. Matthes, E. (2019). Python crash course: A hands-on, project-based introduction to programming (2nd ed.). No Starch Press.
6. Ramalho, L. (2015). Fluent Python: Clear, concise, and effective programming. O'Reilly Media.

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Software Engineering (THEORY)**		
Course Code:	**BCAAIMS403**	No. of Credits	**05**
Contact hours	**60 Hours**	Duration of SEE/Exam	**3 Hours**
Formative Assessment Marks	**20**	Summative Assessment Marks	**80**

Course Outcomes (COs):

After the successful completion of the course, the student will be able to:

- How to apply the software engineering lifecycle by demonstrating competence in communication, planning, analysis, design, construction, and deployment.
- An ability to work in one or more significant application domains.
- Work as an individual and as part of a multidisciplinary team to develop and deliver quality software.
- Demonstrate an understanding of and apply current theories, models, and techniques that provide a basis for the software lifecycle.
- Demonstrate an ability to use the techniques and tools necessary for engineering practice.

Unit	Description	Hours
1	**Introduction:** The Software Problem, Software Engineering Problem, The Software Engineering Approach. **Software Processes:** Software Process, Characteristics of a Software Process, Software Development Process, Waterfall Model, Prototyping, Iterative Enhancement, Spiral Model, Project Management Process, Phases of management process, Metrics, Measurement, and Models, Software Configuration Management Process, Configuration Identification, change control, Status accounting and auditing, Process Management Process, Building estimation models, Process Improvement and maturity.	**15**
2	**Software Requirements Analysis and Specification:** Software Requirements, Need for SRS, Requirement process, Problem Analysis, Analysis Issues, Informal Approach, Structured Analysis, Prototyping, Requirements Specification, Characteristics of an SRS, Components of an SRS, Specification Languages, Structure of a Requirements Document, Validation, Requirement Reviews. **System Design**: Design Principles, Module-Level Concepts, Design Notation and Specification, Data Flow Diagrams, Structured Design Methodology, Verification.	**15**
3	**Detailed Design:** Module specification, specifying functional module, Detailed design, PDL, Logic/Algorithm Design, Verification, Design Walkthroughs, Critical Design Reviews, Consistency checkers. **Coding:** Programming Practice, Top-Down and Bottom-Up, Structured Programming, Information Hiding, Programming Style, Internal	**15**

	Documentation, Verification, Code Reading, Static Analyses, Symbolic Execution, Proving Correctness, Code Inspections or Reviews, Unit Testing.	
4	**Testing and Maintenance:** Testing Fundamentals, Error, Fault, and Failure, Test Oracles, Top-Down and Bottom-Up Approaches, Test Cases and Test Criteria, Psychology of Testing, Functional Testing, Equivalence class partitioning, Boundary value analysis, Cause effect graphing, Structural Testing, Control flow-based criteria, Data flow-based testing, Preventive and Corrective Maintenance. **Introduction to Testing tools:** Features of Test tools, Guidelines for selecting a tool, Tools and skills of testers, Types of testing tools, Difficulties while introducing new tools, Process of procurement of COTS.	**15**

Text Books:

1. Pankaj Jalote, **An Integrated Approach to Software Engineering**, 2nd Edition, Narosa Publishing House, 2004
2. M G Limaye, **Software Testing- Principles, Techniques and Tools**, McGraw Hill Education, 2009

Reference Books:

1. Waman S Jawadekar-Software Engineering Principles and Practice,Tata McGrawHill,2004
2. Roger S. Pressman, **Software Engineering: A Practioner's Approach**, McGraw Hill, 2009

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Artificial Intelligence - I Lab**		
Course Code:	**BCAAIMPS404**	No. of Credits	**02**
Contact hours	**52 Hours**	Duration of SEE/Exam	**3 Hours**
Formative Assessment Marks	**10**	Summative Assessment Marks	**40**

Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	**PART-A** **Writing: 5 Marks Execution: 3 Marks**	**15 Marks**
Program-2	**PART-B** **Writing: 8 Marks Execution: 4 Marks**	**20 Marks**
Practical Record		**05 Marks**
Total		**40 Marks**

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Python Programming Lab**		
Course Code:	**BCAAIMPS405**	No. of Credits	**02**
Contact hours	**52 Hours**	Duration of SEE/Exam	**3 Hours**
Formative Assessment Marks	**10**	Summative Assessment Marks	**40**

PART-A

1. Write a program create list with N elements. find all unique elements in the list. If an element is found only once in the list, then add that element to the unique list.
2. Write a Program, using user-defined functions to find the area of rectangle, square, circle and triangle by accepting suitable input parameters from user.
3. Consider a tuple t1= (1,2,5,7,9,2,4,6,8,10). Write a program to perform following operations:
    a) Print half the values of tuple in one line and the other half in the next line.
    b) Print another tuple whose values are even numbers in the given tuple.
    c) Concatenate a tuple t2= (11,13,15) with t1.
    d) Return maximum and minimum value from this tuple.

4. Write a function that takes a sentence as input from the user and calculates the frequency of each letter. Use a variable of dictionary type to maintain the count.
5. Write a program to create a text file and compute the number of characters, words and lines in a file.
6. Program using user defined exception class that will ask the user to enter a number until he guesses a stored number correctly. To help them figure it out, a hint is provided whether their guess is greater than or less than the stored number using user defined exceptions.
7. Write a Pandas program to create and display a DataFrame from a specified dictionary data which has the index labels.
   Sample Python dictionary data and list labels: exam_data = {'name': ['Asha', 'Disha', 'Risha', 'Sasha', 'Nisha', 'Prisha', 'Usha', 'Raasha', 'Felisha', 'Jisha'], 'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19], 'attempts': [1, 3, 2, 3, 2, 3, 1, 1, 2, 1], 'qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes']} labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

    a) Write a Pandas program to display a summary of the basic information about the specified DataFrame and its data
    b) Write a Pandas program to select the rows where the number of attempts in the examination is greater than 2

PART-B

1. Program to create a class Employee with empno, name, depname, designation, age and salary and perform the following function
    i) Accept details of N employees
    ii) Search given employee using empno
    iii) Display employee details in neat format

2. Write a program menu driven to create a BankAccount class. class should support the following methods for
    i) Deposit
    ii) Withdraw
    iii) GetBalanace .

   Create a subclass SavingsAccount class that behaves just like a BankAccount, but also has an interest rate and a method that increases the balance by the appropriate amount of interest.

3. Write a GUI program to implement Simple Calculator

4. Create a table student table (regno, name and marks in 3 subjects) using MySQL and perform the followings
    a. To accept the details of students and store it in database.
    b. To display the details of all the students
    c. Delete particular student record using regno.

5. Create a table employee (empno, name and salary) using MySQL and perform the followings
    a) To accept the details of employees and store it in database.
    b) To display the details of a specific employee
    c) To display employee details whose salary lies within a certain range

6. Create a table electricity_bill(TariffCode, Customer_Name, Meter Number, Previous_Reading and Current_Reading) using MySQL and perform the Following:
    a) To accept the details of customers and store it in database.
    b) To Update the customer details by Meter Number.
    c) Calculate Bill of a particular customer using below criteria.

Tariff Code	Units Consumed	Rate/Unit
LT1	0-30	2.0
	31-100	3.5
	101-200	4.5
	Above 200	5.0

LT2	0-30	3.5
	31-100	5.0
	101-200	6.0
	Above 200	7.5

7. Consider following data and draw the bar graph using matplot library.(Use CSV or Excel).Add the data Using GUI.

Batsman	**2017**	**2018**	**2019**	**2020**
Virat Kohli	2501	1855	2203	1223
Steve Smith	2340	2250	2003	1153
Babar Azam	1750	2147	1896	1008
Rohit Sharma	1463	1985	1854	1638
Kane Williamson	1256	1785	1874	1974
Jos Butler	1125	1853	1769	1436

Display appropriate title for axis and chart. Also show legends.

Evaluation Scheme for Lab Examination:

Assessment Criteria		
Program-1	**PART-A** **Writing: 5 Marks Execution: 3 Marks**	**15 Marks**
Program-2	**PART-B** **Writing: 8 Marks Execution: 4 Marks**	**20 Marks**
Practical Record		**05 Marks**
Total		**40 Marks**

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Cloud Computing (ELECTIVE)**		
Course Code:	**BCAAIMES401**	No. of Credits	**02**
Contact hours	**26 Hours**	Duration of SEE/Exam	**2 Hours**
Formative Assessment Marks	**10**	Summative Assessment Marks	**40**

Course Outcomes (COs):

- After the successful completion of the course, the student will be able to:
- Explain the core concepts of the cloud computing paradigm such as how and why this paradigm shift came about, the characteristics, advantages and challenges brought about by the various models and services in cloud computing.
- Apply the fundamental concepts in data centres to understand the trade-offs in power, efficiency and cost.
- Identify resource management fundamentals like resource abstraction, sharing and sandboxing and outline their role in managing infrastructure in cloud computing.
- Analyze various cloud programming models and apply them to solve problems on the cloud.

Unit	Description	Hours
1	**Introduction:** Different Computing Paradigms- Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing etc., Comparison of various Computing Technologies; Cloud Computing Basics- What is Cloud Computing? History, Characteristic Features, Advantages and Disadvantages, and Applications of Cloud Computing; Trends in Cloud Computing; Leading Cloud Platform Service Providers.	**8**
2	**Cloud Architecture:** Cloud Service Models- Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS), Comparison of different Service Models; Cloud Deployment Models- Public Cloud; Private Cloud, Hybrid Cloud, Community Cloud; Cloud Computing Architecture- Layered Architecture of Cloud. **Virtualization**-Definition, Features of Virtualization; Types Of Virtualizations-Hardware Virtualization, Server Virtualization, Application Virtualization, Storage Virtualization, Operating System Virtualization; Pros and Cons of Virtualization	**8**
3	**Cloud Application Programming and the Aneka Platform**: Aneka Cloud Application Platform- Framework Overview, Anatomy of the Aneka Container. **Cloud Applications**: Scientific Applications- Healthcare (ECG Analysis in the Cloud) Biology (Protein Structure Prediction and Gene Expression Data Analysis for Cancer Diagnosis), Geoscience (Satellite Image Processing);	**10**

Textbooks:

1. Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi: "Mastering Cloud Computing Foundations and Applications Programming", Elsevier, 2013

Reference Books:

1. Barrie Sosinsky:"Cloud Computing Bible", Wiley- India, 2010
2. K Chandrashekaran: "Essentials of Cloud Computing", CRCPress, 2015
3. Derrick Rountree, Ileana Castrillo:"The Basics of Cloud Computing", Elsevier, 2014

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Object Oriented Analysis and Design(ELECTIVE)**		
Course Code:	**BCAAIMES402**	No. of Credits	**02**
Contact hours	**26 Hours**	Duration of SEE/Exam	**2 Hours**
Formative Assessment Marks	**10**	Summative Assessment Marks	**40**

Course Outcomes (COs):

After the successful completion of the course, the student will be able to:

- Explain the principles and requirements of OOA and Design
- Describe the object-oriented approach to system development, modeling objects, relationships and interactions.
- Analyze Objects and Classes of the software system.
- Construct object model using object types, attributes, structures and associations.
- Analyze Functional and Dynamic Modeling

Unit	Description	Hours
1	**Introduction:** Object orientation concept, OO development concept - Modeling concept, OO methodology, three methods, OO Themes - Abstraction, Encapsulation, combining data and behavior, sharing, Emphasis on the essence of an Object, Synergy **Modeling as a design Technique:** Modelling, Abstraction, The three models **Class modelling:** Object and class concepts - Objects, Classes, Class diagram, Values and attributes, Operation and methods, Link and Association Concepts - Link and association, Multiplicity, Association and names, Ordering, Bags and Sequences, Association Class, Qualified Association, Generalization and Inheritance- Definition, Use of generalization, Overriding features	**8**
2	**State Modelling :**Events - Signal event, change event, Time event, States, Transistors and conditions State Diagrams - Sample State Diagram, one shot state Diagrams, Summary of Basic state diagram notations, State Diagram Behavior - Activity Effects, Do Activities, Entry and Exit Activities, Completion Transition, Sending Signals **Sequence Model**: Scenarios, Sequence Diagram, Communication Diagram, Activity Model - Activities, Branches, Introduction and termination, Concurrent Activities, Executable Activity diagram, Guidelines for Activity models, Deployment Diagram	**8**
3	**Class Design** Overview of Class Design, Bridging the Gap, Realizing Use Cases, Designing Algorithms - Choosing Algorithms, Choosing Data structures, Defining Internal classes and Operations, Assigning Operations to Classes, Recursing Downward - Functionality Layers, Mechanism Layers, Refactoring, Design Optimization - Adding Redundant associations for Efficient Access, Saving derived values to avoid Re-computation, Rectification of Behavior, Adjustment of Inheritance - Rearranging Classes and Operations, Abstracting out Common Behavior, Using Delegation to share Behavior	**10**

Text Book

1. Object Oriented Modeling and Design with UML Michael R. Blaha James R. Rumbaugh, Second Edition, Pearson

Reference Books

1. UML™ 2 Toolkit – Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado, WILEY Publishing
2. Object Oriented Analysis and Design with Applications Grady Booch Second Edition (Pearson Education)
3. Object Oriented Software Engineering Bernd Brugge and Allen H. Dutoit Pearson Education

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Digital Image Processing (ELECTIVE)**		
Course Code:	**BCAAIMES403**	No. of Credits	**02**
Contact hours	**26 Hours**	Duration of SEE/Exam	**2 Hours**
Formative Assessment Marks	**10**	Summative Assessment Marks	**40**

Course Outcomes (COs):

After the successful completion of the course, the student will be able to:

- Remember the fundamental concepts of Image Processing
- Explain different Image enhancement techniques
- Understand and review image transforms
- Analyse and evaluate digital images.

Unit	Description	Hours
1	**Introduction:** What is Digital image processing, The origin of DIP, Examples of fields that use DIP, Fundamentals steps in DIP, Components of an image processing system. **Digital Image Fundamentals:** Elements of Visual perception, Light and the electromagnetic spectrum, Image sensing and acquisition, Image sampling and Quantization, Some Basic relationship between Pixels	**8**
2.	**Image Enhancement in the Spatial Domain:** Background, some basic Gray Level Transformations, Histogram Processing, Enhancement using Arithmetic / Logic operations, Basics of spatial filtering, Smoothing Spatial Filters, Sharpening spatial filters.	**8**
3	**Color Image Processing:** Color Fundamentals, Colour Models, Pseudocolor Image Processing, Colour transformations, Smoothing and Sharpening, Color Segmentation, Noise in Color Images	**10**

Text Books:

1. Rafael C. Gonzalez, Richard E. Woods, "Digital Image Processing", Second Edition, PHI/Pearson

Reference Books:

1. B. Chanda, D. Dutta Majumder, "Digital Image Processing and Analysis", PHI, 2003.
2. Nick Efford, "Digital Image Processing a practical introducing using JAVA", Pearson Education, 2004.
3. Kothari, Ashish M, Digital Image Processing using SCILAB, Springer publication, 2019.

Program Name	**BCA-AIML**	Semester	**IV**
Course Title	**Basic Web Designing Skills (Compulsory)**		
Course Code:	**BCAAIMSS401**	No. of Credits	**02**
Contact hours	**26 Hours**	Duration of SEE/Exam	**2 Hours**
Formative Assessment Marks	**10**	Summative Assessment Marks	**40**

Course Outcomes (COs):

- After completing this course satisfactorily, a student will be able to:
- Understand the fundamentals of HTML5 and its evolution from previous versions.
- Identify the structure and components of an HTML5 document.
- Utilize HTML5 semantic elements to create well-structured web pages..

Unit	Description	Hours
1	**Introduction to Computers and the Internet**-Introduction, The Internet in Industry and Research,HTML5, CSS3,Demos,Evolution of the Internet and World Wide Web, Web Basics. **Introduction to HTML5:**Introduction,Editing HTML5,First HTML5 Example,W3C HTML5 Validation Service, Headings, Linking, Images, Special Characters and Horizontal Rules, Lists, Tables, Forms, Internal Linking ,meta-Elements. **New HTML5 Form input Types**, input and Datalist Elements and autocomplete Attribute, Page-Structure Elements.	10
2	**Cascading Style Sheets**-Introducing CSS, Where You Can Add CSS Rules, CSS Properties-Controlling Fonts, Text Formatting, Text Pseudo-Classes, Selectors, Lengths, Percentages.	8
3	**More Cascading Style Sheets:-**Links, Backgrounds, Lists, Tables, Outlines.	8

Text Books:

1. Deitel, Paul_Deitel, Harvey_Deitel, Abbey - Internet and World Wide Web How to Program-Pearson Education (US) (2011)
2. Jon Duckett -Beginning Web Programming with HTML, XHTML, and CSS (Wrox Beginning Guides)-Wrox (2004)

Reference Books:

1. The Complete Reference HTML and CSS, 5th Edition, Thomas A Powell, 2017.
2. Animation in HTML, CSS, and JAVA Script, Kirupa Chinnathambi, 1st Edition, CreateSpace Independent Pub, 2013

Questions Paper for Pattern Core Subjects

Duration: 3 Hours Max. Marks: 80

Note: Answer any TEN Questions from Part-A. And one full Questions from each unit in Part-B

Part-A

1. 10*2=20
   a.
   b.
   c.
   d.
   e.
   f.
   g.
   h.
   i.
   j.
   k.
   l.

PART B

Each unit contain two main questions, and it carry 10 Marks. Each main questions contain 2 or more sub question.

3*10=30

UNIT-I

2.
   a.
   b.
   c.

3.
   a.
   b.

UNIT-II

4.
   a.
   b.
   c.

5.
   a.
   b.
   c.

UNIT-III

6.
   a.
   b.
   c.

7.
   a.

b.
c.

UNIT-IV

8.
a.
b.
c.

9.
a.
b.

Questions Paper Pattern for Elective and Compulsory Subjects

Duration:2Hours Max.Marks:40

Note: Answer any 5 Questions from Part-A. And one full Questions from each unit in Part-B

Note: Answer any Five Questions from Part-A. And any six Questions out of Nine Questions from Part-B

Part-A

1. 5*2=10
a.
b.
c.
d.

e.
f.
g.
h.

Part-B

Answer any Six questions out of Nine questions.

6*5=30

2.
3.
4.
5.
6.
7.
8.
9.
10.
