



**UNIVERSITY OF RAJASTHAN, JAIPUR**

**NATIONAL EDUCATION POLICY-2020**

**PROPOSED STRUCTURE OF UG-BACHELOR OF  
COMPUTER APPLICATIONS (BCA)**

**(Programme Code: UG0801)**

**Syllabus for 1<sup>st</sup> to 6<sup>th</sup> Semesters**

**and**

**Open Elective Courses in Computer Application**

**SYLLABUS 2023**

**Faculty of Science**

**BCA Part First(I & II Sem) -2023-24**

**BCA Part Second(III & IV Sem) -2024-25**

**BCA Part Third(V & VI Sem) -2025-26**

*Rj | Jau*  
Dy. Registrar  
(Academic)  
University of Rajasthan  
JAIPUR

## Preamble

Computer Application (CA) has been evolving as an important branch of science and technology in last three decade and it has carved out a space for itself like computer science and engineering. Computer application spans theory and more application and it requires thinking both in abstract terms and in concrete terms.

The ever -evolving discipline of computer application has strong connections to other disciplines. Many problems in science, engineering, health care, business, and other areas can be solved effectively with computers and its applications, but finding a solution requires both computer science expertise and knowledge of the particular application domain.

Computer science has a wide range of specialties. These include Computer Architecture, Software Systems, Graphics, Artificial Intelligence, Mathematical and Statistical Analysis, Data Science, Computational Science, and Software Engineering.

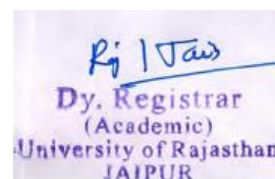
Universities and other HEIs introduced programmes of computer application. Information Technology is growing rapidly. Increasing applications of computers in almost all areas of human endeavour has led to vibrant industries with concurrent rapid change in technology. Unlike other basic disciplines, developing core competency in this discipline that can be reasonably stable becomes a challenge.

In India, it was initially introduced at the Master (postgraduate) level as MCA and M.Tech. Later on, engineering programmes such as B.Tech and B.E in Computer Science & Engineering and in Information Technology were introduced in various engineering College/Institutions to cater to the growing demand for trained engineering manpower in IT industries. Parallely, BCA, BSc and MSc programmes with specialisation in Computer Science were introduced to train manpower in this highly demanding area.

BCA and BCA (Hons) are aimed at undergraduate level training facilitating multiple career paths. Students so graduated, can take up postgraduate programmes in CS or MCA leading to research as well as R&D, can be employable at IT industries, or can pursue a teaching profession or can adopt a business management career.

BCA and BCA (Hons) aims at laying a strong foundation of computer application at an early stage of the career. There are several employment opportunities and after successful completion of BCA, graduating students can fetch employment directly in companies as programmer, Web Developer, Software Engineer, Network Administrator, Data Scientist, or AI/ML personnel.

The Program outcomes in BCA are aimed at allowing flexibility and innovation in design and development of course content, in method of imparting training, in teaching learning process and in assessment procedures of the learning outcomes. The emphasis in BCA courses, in outcome-based curriculum framework, help students learn solving problems, accomplishing IT tasks, and expressing

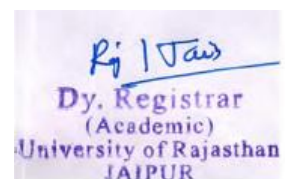


creativity, both individually and collaboratively. The proposed framework will help Students learn programming techniques and the syntax of one or more programming languages.

All students must, therefore, have access to a computer with a modern programming language installed. The computer science framework does not prescribe a specific language. The teacher and students will decide which modern programming languages students will learn. More importantly, students will learn to adapt to changes in programming languages and learn new languages as they are developed.

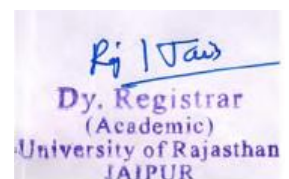
The present Curriculum Framework for BCA degrees is intended to facilitate the students to achieve the following.

- To develop an understanding and knowledge of the basic theory of Computer Science and Information Technology with good foundation on theory, systems and applications such as algorithms, data structures, data handling, data communication and computation
- To develop the ability to use this knowledge to analyse new situations in the application domain
- To acquire necessary and state-of-the-art skills to take up industry challenges. The objectives and outcomes are carefully designed to suit to the above-mentioned purpose.
- The ability to synthesize the acquired knowledge, understanding and experience for a better and improved comprehension of the real-life problems
- To learn skills and tools like mathematics, statistics and electronics to find the solution, interpret the results and make predictions for the future developments
- To formulate, to model, to design solutions, procedure and to use software tools to solve real world problems and evaluate



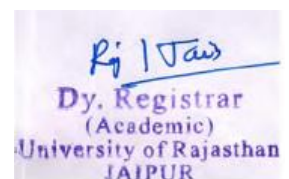
## **The objectives of the Programme are:**

1. The primary objective of this program is to prepare students for careers in software industry, understanding and skills, related to the use of computers and its applications.
2. The course is designed to function as an intermediate between the industry and academic institutes.
3. This course provides students with options to specialize in new and upcoming technologies.
4. To impart creativity and pursuit of excellence in computer applications.
5. To provide opportunity for the study of modern methods of information processing and its applications.
6. To develop among students the programming techniques and the problem solving skills through programming.
7. To develop the ability to use this knowledge to analyze new situations.
8. To be able to blend the acquired knowledge, understanding, and experience, for a better and improved intellectual capacity of the real-life problems.
9. To prepare students who wish to go on to further studies in computer science and related subjects.



## **Program Outcomes: BCA (3 Years) Degree**

1. Discipline knowledge: Acquiring knowledge on basics of Computer Science and ability to apply to design principles in the development of solutions for problems of varying complexity.
2. Problem Solving: Improved reasoning with strong mathematical ability to Identify, formulate and analyze problems related to computer science and exhibiting a sound knowledge on data structures and algorithms.
3. Difficulty Analysis: Talent to classify, significantly evaluate and prepare complex computing problems using fundamentals of computer knowledge and request domains.
4. Design and Development of Solutions: Ability to design and development of algorithmic solutions to real world problems and acquiring a minimum knowledge on statistics and optimization problems. Establishing excellent skills in applying various design strategies for solving complex problems.
5. Accomplish Investigations of Compound Computing Troubles: Ability to invent and ways experiments interpret data and present well up to date conclusions.
6. Application Systems Knowledge: Possessing a sound knowledge on computer application software and ability to design and develop app for applicative problems.
7. Modern Tool Usage: Identify, select and use a modern scientific and IT tool or technique for modeling, prediction, data analysis and solving problems in the area of Computer Science and making them mobile based application software.
8. Mission Administration: Skill to recognize administration and computing philosophy with computing acquaintance to supervise projects in multidisciplinary environments.
9. Communication: Must have a reasonably good communication knowledge both in oral and writing.
10. Ethics on Profession, Environment and Society: Exhibiting professional ethics to maintain the integrality in a working environment and also have concern on societal impacts due to computer-based solutions for problems.
11. Motivation to take up Higher Studies: Inspiration to continue educations towards advanced studies on Computer Science.



# Course Structure for BCA

## BCA Part - I (I & II Sem) 2023-24 Onwards

Semester-wise Titles of the Papers in Bachelor of Computer Application						
Certificate in Computer Application						
Sem	Course Code	Course Title	Course Type	Theory/ Practical	Teaching Hours/ Week	Credits
I	BCA-51T-101	Programming in C	CC(Major)	Theory	4	4
	BCA-51P-102	Programming in C Lab	CC(Major)	Practical	4	2
	BCA-51T-103	Web Application Development	CC(Major)	Theory	4	4
	BCA-51P-104	Web Application Development Lab	CC(Major)	Practical	4	2
	BCA-51T-105	Computer Fundamentals& Office Management Tools	CC(Major/ Minor)	Theory	4	4
	BCA-51P-106	Office Management Tools Lab	CC(Major/ Minor)	Practical	4	2
	BCA-51T-107	AEC1	AEC	Theory	4	4
	BCA-51T-108	SEC1	SEC	Theory	2	2
	BCA-51T-109	VAC1	VAC	Theory	2	2
				<b>Total</b>		<b>32</b>
II	BCA-52T-111	Operating Systems	CC(Major)	Theory	4	4
	BCA-52P-112	Operating Systems Lab	CC(Major)	Practical	4	2
	BCA-52T-113	Database Management Systems	CC(Major)	Theory	4	4
	BCA-52P-114	DBMS Lab	CC(Major)	Practical	4	2
	BCA-52T-115	Computer Organization & Architecture	CC(Major/ Minor)	Theory	6	6
	BCA-52T-116	AEC2	AEC	Theory	4	4
	BCA-52P-117	SEC2	SEC	Practical	4	2
	BCA-52T-118	VAC2	VAC	Theory	2	2
				<b>Total</b>		<b>32</b>

  
 Dy. Registrar  
 (Academic)  
 University of Rajasthan  
 JAIPUR

## BCA Part - II ( III & IV Sem) 2024-25 Onwards

Semester-wise Titles of the Papers in Bachelor of Computer Application						
Diploma in Computer Application						
Semester	Course Code	Course Title	Course Type	Theory/ Practical	Teaching Hours/ Week	Credits
III	BCA-63T-201	Data Structures and Algorithms	CC(Major)	Theory	4	4
	BCA-63P-202	Data Structures Lab Using C	CC(Major)	Practical	4	2
	BCA-63T-203	Object Oriented Programming Through C++	CC(Major)	Theory	4	4
	BCA-63P-204	OOP Lab	CC(Major)	Practical	4	2
	BCA-63T-205	Software Engineering	CC(Major/ Minor)	Theory	6	6
	BCA-63T-206	MEC1	MEC	Theory	4	4
	BCA-63P-207	SEC3	SEC	Practical	4	2
	BCA-63T-208	VAC3	VAC	Theory	2	2
				<b>Total</b>		<b>32</b>
IV	BCA-64T-211	PHP Programming	CC(Major)	Theory	4	4
	BCA-64P-212	PHP Lab	CC(Major)	Practical	4	2
	BCA-64T-213	Object Oriented Concepts Using Java Programming	CC(Major)	Theory	4	4
	BCA-64P-214	Java Lab	CC(Major)	Practical	4	2
	BCA-64T-215	Mathematics & Statistics	CC(Major/ Minor)	Theory	6	6
	BCA-64T-216	MEC2	MEC	Theory	4	4
	BCA-64T-217	SEC4	SEC	Theory	2	2
	BCA-64T-218	VAC4	VAC	Theory	2	2
				<b>Total</b>		<b>30</b>

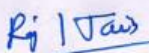
  
 Dy. Registrar  
 (Academic)  
 University of Rajasthan  
 JAIPUR

*Raj / Jas*  
Dy. Registrar  
(Academic)  
University of Rajasthan  
JAIPUR



## BCA Part - III (V & VI Sem) 2025-26 Onwards

Semester-wise Titles of the Papers in Bachelor of Computer Application						
Bachelor of Computer Application						
Semester	Course Code	Course Title	Course Type	Theory/ Practical	Teaching Hours/ Week	Credits
V	BCA-75T-301	Artificial Intelligence & Machine Learning	CC(Major)	Theory	4	4
	BCA-75P-302	Machine Learning Lab	CC(Major)	Practical	4	2
	BCA-75T-303	Python Programming	CC(Major)	Theory	4	4
	BCA-75P-304	Python Lab	CC(Major)	Practical	4	2
	BCA-75T-305	Data Communication & Computer Networks	DSE(Major /Minor)	Theory	6	6
	BCA-75T-306	MEC3	MEC	Theory	4	4
			<b>Total</b>		<b>26</b>	<b>22</b>
VI	BCA-76T-311	.NET Framework with C#	CC(Major)	Theory	4	4
	BCA-76P-312	.NET with C# Lab	CC(Major)	Practical	4	2
	BCA-76T-313	Introduction to Data Science	CC(Major)	Theory	4	4
	BCA-76P-314	Data Science Lab	CC(Major)	Practical	4	2
	BCA-76T-315	Cloud Computing	DSE(Major /Minor)	Theory	6	6
	BCA-76T-316	SEC5	SEC	Theory	2	2
			<b>Total</b>		<b>24</b>	<b>20</b>

  
 Dy. Registrar  
 (Academic)  
 University of Rajasthan  
 JAIPUR

## List of Multidisciplinary Elective Courses(MEC) in Bachelor of Computer Application

S.No.	Course Code	Course Title	Theory/ Practical	Teaching Hours/ Week	Credits	To be Opted in the Semester
1	BCAMEC1	C Programming Concepts	Theory	4	4	III
2	BCAMEC2	Web Application Development	Theory	4	4	III
3	BCAMEC3	Fundamentals of Computer	Theory	4	4	III
4	BCAMEC4	Office Management Tools	Theory	4	4	III
5	BCAMEC5	E-Commerce Technologies	Theory	4	4	IV
6	BCAMEC6	Programming in Python	Theory	4	4	IV
7	BCAMEC7	Digital Marketing	Theory	4	4	IV
8	BCAMEC8	Computer Accounting	Theory	4	4	IV
9	BCAMEC9	Multimedia and Animation	Theory	4	4	V
10	BCAMEC10	Introduction to Cyber Security	Theory	4	4	V
11	BCAMEC11	Open Source with PHP	Theory	4	4	V
12	BCAMEC12	Graphic Design	Theory	4	4	V

*Rj / Jau*  
Dy. Registrar  
(Academic)  
University of Rajasthan  
JAIPUR

## Course Content for BCA, Semesters I and II 2023-24

### Semester: I

Course Code: <b>BCA-51T-101</b>	Course Title: <b>Programming in C</b>
Course Credit : 04	Hours/Week: 04

### Course Objectives (COs):

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

- To Understand the basic language implementation techniques
- Develop ability to learn new languages more quickly
- To understand the concept of functional programming language
- Develop ability to learn and write small programs in different programming Languages

### Course Outcomes (COs):

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

- Understand the basic programming concepts and syntax of the C language.
- Design and implement algorithms to solve simple programming problems.
- Read, understand and trace the execution of programs written in C language
- Write the C code for a given problem
- Perform input and output operations using programs in C
- Write programs that perform operations on arrays Course Content
- Write, compile, and execute C programs using integrated development environments (IDEs).

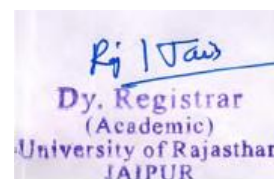
### **BCA-51T-101: Programming in C**

#### **UNIT- I**

Basic concepts of Programming languages, Programming Domains, Language Evaluation criteria and language categories, Evolution of major programming languages. Describing syntax and semantics, formal methods of describing syntax, Pseudo code, Design of Algorithm & Flowchart

#### **UNIT-II**

**Fundamentals of C:** History and importance of C, basic structure and execution of C programs, constants, variables, and data types, Various type of declarations, operators types and expressions,



evaluation of expressions, operator precedence and associability. Managing input and output operations, decision making and branching.

**Iteration:** while, do...while, for loop, nested loops, break & continue, goto statements.

### UNIT-III

**Array and String:** One-dimensional array and their declaration and initialization, two-dimensional arrays and their initializations, character arrays (One and Two dimensional), reading and writing strings, string - handling functions.

**Functions:** Need and elements for user –defined functions, definition of functions, return values and their types, function calls and declaration, recursion, parameter passing, passing arrays and strings to functions, the scope, visibility and life time of variables.

### UNIT-IV

**Understanding Pointers:** Accessing the address of a variable, declaration and initialization of pointer variables, accessing a variable through its pointer, pointers and arrays, pointers and function arguments, functions returning pointers.

**Structures and Unions:** Defining structure, declaring structure variable and accessing structure members, initialization of structure, operation on individual members, and array of structures, union, size of structure.

#### Recommended Books:

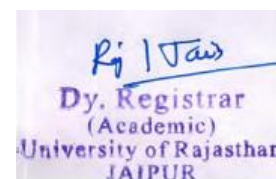
1. Balagurusamy E; Programming in ANSI C;FifthEdn; Mc Graw Hill,2011.
2. KanetkarY.; LET US C; X Edition, BPB,2010.
3. Deitel HM & Deitel JP; C How to program; 5<sup>th</sup>Edn; Pearson Pub
4. GottfriedB; Programming with C: SchaumQutlines; Mc Graw Hill Edition.

Corse Code: <b>BCA-51P-102</b>	Course Title: <b>Programming in C Lab</b>
Course Credit : 02	Hours/Week: 04

Content : **Recommended exercises**

Part A:

1. Program to read radius of a circle and to find area and circumference
2. Program to read three numbers and find the biggest of three
3. Program to demonstrate library functions in math.h
4. Program to check for prime
5. Program to generate n primes
6. Program to read a number, find the sum of the digits, reverse the number and check it for palindrome



7. Program to read numbers from keyboard continuously till the user presses 999 and to find the sum of only positive numbers
8. Program to read percentage of marks and to display appropriate message (Demonstration of else-if ladder)
9. Program to find the roots of quadratic equation (demonstration of switch Case statement)
10. Program to read marks scored by n students and find the average of marks (Demonstration of single dimensional array)
11. Program to remove Duplicate Element in a single dimensional Array
12. Program to perform addition and subtraction of Matrices

**Part B:**

1. Program to find the length of a string without using built in function
2. Program to demonstrate string functions.
3. Program to demonstrate pointers in C
4. Program to check a number for prime by defining isprime( ) function
5. Program to read, display and to find the trace of a square matrix
6. Program to read, display and add two m x n matrices using functions
7. Program to read, display and multiply two m x n matrices using functions
8. Program to read a string and to find the number of alphabets, digits, vowels, consonants, spaces and special characters.
9. Program to Reverse a String using Pointer
10. Program to Swap Two Numbers using Pointers
11. Program to demonstrate student structure to read & display records of n students.
12. Program to demonstrate the difference between structure & union.

**Note: Student has to execute a minimum of 10 programs in each part to complete the Lab course.**

Corse Code: <b>BCA-51T-103</b>	Course Title: <b>Web Application Development</b>
Course Credit : <b>04</b>	Hours/Week: <b>04</b>

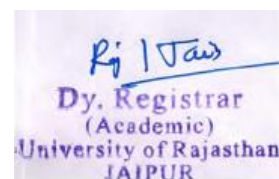
**Course Objectives (COs):**

By the end of this course, students will be able to:

- Comprehend the optimal technologies for addressing web client/server challenges.
- Evaluate and create real-time web applications.
- Utilize JavaScript for dynamic effects and form input validation.
- Analyze and select suitable client-side and server-side application technologies.

**Course Outcomes (COs):**

**On completion of the course, the student will be able to:**



- Understand best technologies for solving web client/server problems
- Analyze and design real time web applications
- Use Java script for dynamic effects and to validate form input entry
- Analyze to Use appropriate client-side and Server-side application technology

## **BCA-51T-103 : Web Application Development**

### **Unit – I**

**The Internet** – Basic of internet, file transfer, telnet, usenet, gopher, wais, Archie and veronica. Introduction to Internet Protocols-, HTTP, FTP, SMTP protocols.

**World Wide Web** : Elements of the Web, Web browser and its architecture, The web server, the proxy server, Microsoft internet explorer, viewing pages with a browser, using a browser for Mail, News and chat, Security and Privacy issues (cookies, firewalls, Data Security, executable Applets and scripts, blocking system).

### **Unit – II**

**HTML Fundamentals:** Introduction to HTML, HTML Elements, HTML Semantics, HTML 5 Doc Types, New Structure Tags, Section, Nav, Article, Aside, Header, Footer, HTML Attributes, Headings, Paragraphs, Styles, Quotations, Blocks, Classes, Layout, Iframes, Creating HTML Pages, incorporating Horizontal Rules and Graphical Elements, Hyper-links, Creating HTML Tables, Creating HTML Forms, HTML and Image Techniques, HTML and Page, Development of Website and Webpage (Planning, Navigation and Themes, Elements of a Web page, steps of creating a site, publishing and publicizing site structuring web site.

### **Unit–III**

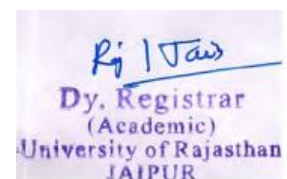
**Cascading Style Sheets:** Understanding Style Sheets, CSS Syntax and Applying Style Sheets to HTML document, Developing Style Sheets: inline, internal and external. CSS Selectors, <DIV> tag, Using class and ID, Styling Backgrounds, Styling borders, Styling Text, Styling Fonts, Styling Links, Styling Lists, Styling Tables, Margin, Flex and Grids. **Bootstrap & Web page design** : CMS, Banks of CMS, Joomla/wordpress-Installation, Design and development of websites.

### **Unit–IV**

**Java script:** Introduction to scripting language, Client Side Scripting, memory concepts, arithmetic decision making. Java script control structures, Java script functions, JS Popup Boxes, events, program modules in java script, function definitions duration of identifiers, scope rules, Controlling Programming Flow, recursion java script global functions. Arrays handling in Java script, The Java Script Object Model, Developing Interactive Forms, Validation of Forms, Cookies and Java Script Security Controlling Frames in Java Script, Client – Side Java Script Custom.

#### **References :**

1. The Colete eference: HTML & XHTML; ThomasA.Powel, 4<sup>th</sup>Edn.



2. Mastering HTML 4.0 by DeborahS.Ray an EricJ.Ray From BPB
3. Mastering Java Script, BPB publication.
4. Internet and web technology by Raj Kamal, TMH Publication 2. StevenHolzner,
5. The Complete Reference Java Scripts,, TataMcGraw – Hill,3<sup>rd</sup>Edn.
6. Java Script, DonGosselin,Vikas publications

Corse Code: <b>BCA-51P-104</b>	Course Title: <b>Web Application Development Lab</b>
Course Credit : <b>02</b>	Hours/Week: <b>04</b>

**Content : Recommended exercises**

**HTML:**

1. Basics Elements & Attributes, HTML Formatting tags, Links,
2. Images, Tables, Forms Elements
3. HTML5 Audio and Video, HTML5 Input Types & Attributes
4. CSS Syntax, CSS Attribute Selectors
5. CSS properties: Fonts, Background, Colors, Links, Lists,
6. CSS Box Model, Display, Opacity, Float, Clear
7. CSS Layout, CSS Navigation Bar,
8. CSS Rounded Corners, CSS Border Images, CSS Animations

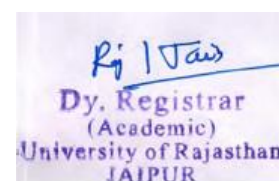
**JavaScript:**

1. Displaying Output, DeclaringVariables, Operators, Arithmetic, Data Types, Assignment,
2. JavaScript Functions, Booleans, Comparisons, Conditional ,
3. JavaScript Switch, Loops, Break, Type,
4. JavaScript Objects, Scope,
5. Strings and String Methods
6. Numbers and Number Methods, Math, JavaScript Dates: Formats and Methods
7. JavaScript Events, JavaScript, JavaScript Forms (API and Validation), Objects,
8. JavaScript Functions, JavaScript DOM, JavaScript Validation, Browser BOM

Corse Code: <b>BCA-51T-105</b>	Course Title: <b>Computer Fundamentals &amp; Office Management Tools</b>
Course Credit : 04	Hours/Week: 04

**Course Objectives (COs):**

By the end of this course, students will be able to:



- Understand Fundamental Computing Concepts and gain a solid foundation in the basic principles of computer science, including hardware, software, and networking.
- Develop Proficiency in Office Management Tools such as word processors, spreadsheets, presentation tools, and database management systems.
- Apply Information Technology in Business Contexts to Understand how information technology can be applied to solve business problems and improve organizational efficiency.

### Course Outcomes (COs):

- Introduction to computers, classification of computers, anatomy of computer, constituents and architecture, microcontrollers
- Internet basics, features, applications, services, internet service providers, domain name system, browsing, email, searching
- Introduction to Internet basic, e-mail, Web basics
- Office Activities using Word Processor Software
- Office Activities using Spreadsheets Software
- Office Activities using Presentation Software
- Office Activities using Database Software
- Office Activities involving Multimedia Editing (Images, Video, Audio ...)
- Operating System Configuration, MS Configuration.

## BCA-51T-105: Computer Fundamentals & Office Management Tools

### UNIT- I

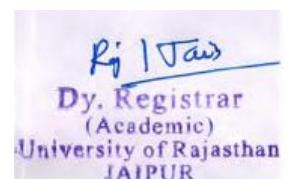
**Introduction to Computers:** Characteristics of computers, Evolution of computers, generation of computers, Block diagram of computer & role of each block, classification of computers. Input and Output Devices

**Primary and Secondary Memory:** Memory hierarchy, Random access memory (RAM), types of RAM, Read only memory (ROM), types of ROM. Classification of secondary storage devices, magnetic tape, magnetic disk, optical disk.

**Number Systems:** Introduction to number system, Binary, Octal, Hexadecimal, conversion between number bases, Arithmetic operations on binary numbers, Alphanumeric- BCD, EBCDIC, ASCII, Unicode.

### UNIT-II

**Computer Software:** software categories, system software, application software, utility software. Classification of system software, **Computer Languages:** Introduction, classification of





programming languages, generations of programming languages, features of a good programming language.

**Internet Basics:** Introduction,, Features of Internet, Internet applications, Services of Internet, Logical and Physical addresses, Internet Service Providers, Domain Name System. **Web Basics :** Introduction to Web, Web browsers, http/https, URL.

### UNIT-III

**MS Word:** Word processing, MS-Word features, creating saving and opening documents in Word, interface, toolbars, ruler, menus, keyboard shortcut, editing, previewing, printing & formatting a document, advance features of MS Word, find & replace, using thesaurus, mail merge, handling graphics, tables, converting a Word document into various formats like-text, rich text format, Word perfect, etc.

**MS Excel:** Worksheet basics, creating worksheet, entering data into worksheet, data, text, dates, alphanumeric values saving & quitting worksheet, opening and moving around in an existing worksheet, Toolbars and menus, Keyboard shortcuts, working with single and multiple workbook, working with formula & cell referencing, Auto sum, coping formulas, absolute and relative addressing, formatting of worksheet, previewing & printing worksheet, Graphs and Charts, Database, macros, multiple worksheets-concepts.

### UNIT-IV

**Power Point:** Creating and viewing a presentation, managing Slide Shows, navigating through a presentation, using hyperlinks, advanced navigation with action setting and action buttons, organizing formats with MasterSlides, applying and modifying designs, adding graphics, multimedia and special effects.

**Microsoft Access:** Planning a database (tables, queries, forms, reports), creating and editing database, customizing tables, linking tables, designing and using forms, modifying database structure, Sorting and Indexing database, querying a database and generating reports.

#### Reference Books:

1. SanjaySaxena; A First Course in Computers 2003 Edition; VikasPub.
2. Computer Fundamentals by P.K.Sinha, BPB Publication.
3. Computer Fundamentals and Programming in C,ReemaThareja,OXFORDUniversity Press.
4. Microsoft; 2007/2010 Microsoft Office System; PHI.
5. Microsoft; Microsoft Office 2007/2010: Plain & Simple; PHI.
6. MS-Office , Dr.S.S.Shrivastava, Published by Laxmi Publication.
7. Office 2019:In Easy Steps,MichalPrice ,BPB Publication.

Corse Code: <b>BCA-51P-106</b>	Course Title: <b>Office Management Tools Lab</b>
--------------------------------	--



Course Credit : 02	Hours/Week: 04
--------------------	----------------

**Content: Content : Recommended exercises**

Exercises based on Word, Excel, Power Point and Access.

*Rj | Jau*  
Dy. Registrar  
(Academic)  
University of Rajasthan  
JAIPUR

## Course Content for BCA, Semesters II

### Semester: II

Course Code: <b>BCA-52T-111</b>	Course Title: <b>Operating Systems</b>
Course Credit : <b>04</b>	Hours/Week: <b>04</b>

### Course Objectives (COs):

Upon finishing the course, students will be able to:

- To design and understand the following OS components: System calls, Schedulers, Memory management systems, Virtual Memory and Paging systems.
- To evaluate, and compare OS components through instrumentation for performance analysis.
- To analyze the various device and resource management techniques for timesharing and distributed systems
- To develop and analyze simple concurrent programs using transactional memory and message passing, and to understand the trade-offs and implementation decisions

### Course Outcomes (COs):

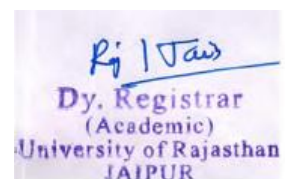
- Understand fundamental operating system abstractions such as processes, threads, files, semaphores, IPC abstractions, shared memory regions, etc.,
- Analyse important algorithms e.g. Process scheduling and memory management algorithms
- Categorize the operating system's resource management techniques, dead lock management techniques, memory management techniques
- Demonstrate the ability to perform System Administration tasks in LINUX

## **BCA-52T-111 : Operating Systems**

### **Unit – I**

**Concepts:** Operation System & its need, functions of OS, Types of OS : Simple Batch Systems, Multiprogrammed Batched Systems, Time-Sharing Systems, Parallel Systems, Distributed Systems and Real-Time Systems.

**Operating-System Structures:** System Components, Operating System Services, System Calls, System Structure, Virtual Machines, Process Management.



## Unit – II

**CPU Scheduling Algorithms** :Basic Concepts, Scheduling Criteria, FCFS, SJF, Priority, Round-Robin, Multilevel Queue, Multilevel Feedback Queue, Multiple-Processor Scheduling.

**Process Synchronization & Deadlocks**: The Critical section problem, synchronization hardware semaphores, Classical problems of synchronization, Critical regions, System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Recovery from Deadlock.

## Unit-III

**Memory Management**: Background, Logical versus Physical Address space, Swapping, Contiguous allocation (fragmentation), Paging, Segmentation. Virtual Memory, Demand Paging, Page-replacement Algorithms (FIFO, Optimal, LRU, Counting).

**File Management**: File Concepts (Operations & Attributes), Access Methods, Directory Structure, File System Structure, Allocation Methods (Contiguous Allocation, Linked Allocation, Indexed Allocation).

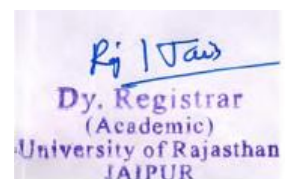
**Device Management**: General device characteristics, device controllers, device drivers, Interrupts Driven I/O, Memory Mapped I/O, Direct Memory.

## Unit-IV

Introduction to Linux , Evolution of Linux, Linux Architecture, Linux file system (inode, Super block, Mounting and Unmounting), Essential Linux Commands and Shell Scripts (Internal and External Commands), Kernel, Process Management in Linux.

### Recommended reference books:

1. A.Silberschatz and P.Galvin, “Operating System Concepts”, Addison-Wesley, 5th Ed., 2001.
2. GaryNutt: Operating Systems-A Modern Perspective (Second Edition), Pearson Education, 2000.
3. Tanenbaum A.S., Modern Operating Systems, PHI Publ.
4. PetersonRichard, “ The Complete Reference Linux “ Tata McGraw Hill.
5. SimitabhaDas, “Unix/Linux Concepts & Applications”. Tata McGraw Hill
6. AchyutS.Godbole: Operating Systems, Tata Mc-Graw Hill Publishing Company Limited, 2000.
7. HarveyM.Deitel, Operating Systems, Pearson Education, 2001.



Corse Code: <b>BCA-52P-112</b>	Course Title: <b>Operating Systems Lab</b>
Course Credit : <b>02</b>	Hours/Week: <b>04</b>

### Course Outcomes (COs):

- Understand fundamental operating system abstractions such as processes, threads, files, semaphores, IPC abstractions, shared memory regions, etc.,
- Analyse important algorithms e.g. Process scheduling and memory management algorithms
- Categorize the operating system's resource management techniques, dead lock management techniques, memory management techniques
- Demonstrate the ability to perform System Administration tasks in LINUX

### Content : Recommended exercises

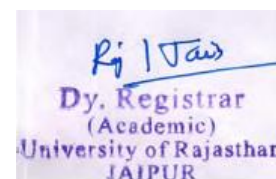
1. Settings and configurations of Linux.
2. To learn directory navigation in Linux-like systems.
3. To practice Linux commands.
4. Practice pattern matching commands.
5. Practice file editing with vi/nano.
6. Shell script to demonstrate application programs.

Corse Code: <b>BCA-52T-113</b>	Course Title: <b>Database Management Systems</b>
Course Credit : 04	Hours/Week: 04

### Course Objectives (COs):

Upon finishing the course, students will be able to:

- Upon finishing the course, students will be able to:
- To study types of MySQL databases (Document oriented, keyValue pairs, Column-oriented and Graph)



- To understand detailed architecture, define objects, load data, query data and performance tune MySQL databases.
- Able to handle large volume of data through queries.

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

- Understand terms related to database design and management
- Assess various database models.
- Evaluate the normality of a logical data model, and correct any anomalies
- Implement relational databases using MySQL.

## **BCA-52T-113 : Database Management Systems**

### **UNIT- I**

**Database System Concepts & Architecture:** Overview of DBMS, Basic DBMS terminology, data base system v/s file system, Advantages and dis-advantages of DBMS, Coded rules, data independence. Architecture of a DBMS, Schemas, Instances, Database Languages, Database Administrator, Data Models.

### **UNIT-II**

**Data Modeling:** Data modeling using the Entity Relationship Model: ER model concepts, notation for ER diagram, mapping constraints, keys, Concepts of Super Key, candidate key, primary key, Generalization, aggregation.

**Relational Model :** Concepts, Constraints, Languages, Relational database design by ER & EER mapping, Relational algebra relational calculus. Relational Algebra, Fundamental operations of Relational Algebra.

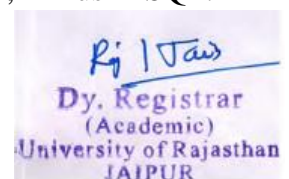
### **UNIT –III**

**Database Design:** Functional dependencies, loss less decomposition, Normalization : 1-NF, 2-NF,3-NF and BCNF. **Transaction Management :** Transactions: Concepts, ACID Properties, States Of Transaction, Serializaibility, Isolation, Checkpoints, Deadlock Handling.

**Recovery System & Security :** Failure Classifications, Recovery & Atomicity, Log Base Recovery, Recovery with Concurrent Transactions, Introduction to Security & Authorization.

### **UNIT-IV**

**Introduction to SQL:** Characteristics of SQL, Advantages of SQL, SQL data types and literals, Types of SQL commands, SQL operators and their procedure, Tables, views and indexes, Queries and sub queries, Aggregate functions, insert, update and delete operations, Joins, Unions, Intersection, Minus in SQL.



### Recommended Books:

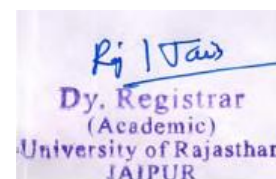
1. KorthHF and SilberschatazA, System Concepts, Sixth Edition; McGraw Hill,2010
2. Leon, and Leon, SQL Tata McGraw Hill Pub. Co. Ltd.
3. IvanBayross; SQL/PL 4<sup>th</sup>Edn: BPB,2009
4. NavatheS.B.ElmasriR,; Fundamentals of Database Systems, Fifth Edition, Pearson 2011.
5. Ramakrishan and Gharke, Database Management Systems, 3<sup>rd</sup>Ed, Tata McGraw Hill, 2007.
6. Singh S.K.; Database Systems; I Edition; Pearson, 2006.

Corse Code: <b>BCA-52P-114</b>	Course Title: <b>DBMS Lab</b>
Course Credit : 02	Hours/Week: 04

### Course Contents : Recommended exercises

1. Analyze the organization and identify the entities, attributes and relationships in it.
2. Identify the primary keys for all the entities. Identify the other keys like candidate keys, partial keys, if any.
3. Relate the entities appropriately. Apply cardinalities for each relationship. Identify strong entities and weak entities (if any).
4. Represent all the entities (Strong, Weak) in tabular fashion. Represent relationships in a tabular fashion.
5. Apply the First, Second and Third Normalization levels on the database designed for the organization
6. Practicing DDL commands.
7. Creating databases, how to create tables, altering the database, dropping tables and databases if not required. Try truncate, rename commands etc.
8. Practicing DML commands on the Database created for the example organization
9. DML commands are used to for managing data within schema objects. Some examples: SELECT, INSERT, UPDATE, DELETE
10. Practice queries (along with sub queries) involving ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, Constraints etc.
11. Practice queries using Aggregate functions (COUNT, SUM, AVG, and MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.

Corse Code: <b>BCA-52T-115</b>	Course Title: <b>Computer Organization &amp; Architecture</b>
Course Credit : 06	Hours/Week: 06



## Course Objectives (COs):

Upon finishing the course, students will be able to:

- The students will be able to understand digital logic design, including logic elements, and their use in combinational and sequential logic circuit design, the basic architecture of processing, memory and I/O organization in a computer system.
- Understand the design of Sequential and Arithmetic Circuits.
- Understand the functions of CPU and I/O devices.
- Understand the operations and structure of Memory.

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

- Understand Boolean Algebra and Data Representation.
- Understand and Design of Sequential and Arithmetic Circuits.
- Boolean Algebra and Data Representation.
- Understand Microprocessor operations.
- Remember and Understand the basics of computer architecture, organization and Design.
- Understand the operations of CPU and I/O devices.
- Understand the operations and organization of Memory.
- Understand the concept of parallel processing and pipelining

## BCA-52T-115 : Computer Organization & Architecture

### UNIT- I

**Boolean Algebra and Logic Gates:** Logic Gates, Basic laws of Boolean algebra, Simplification of Boolean algebra.

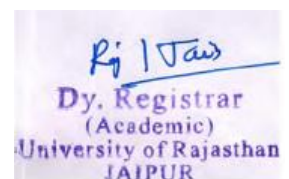
**Combinatorial Logic :** Multiplexers, Decoders, Encoders, Adder & Subtracters, Parallel Binary Adder, Parallel binary Subtractor .

### UNIT-II

**Sequential Logic:** Sequential circuits: Flip-flops, S-R, D, J-K, T, Clocked Flip-flop, Race around condition, Master slave Flip-Flop.

**Register Transfer and Micro Operations:** Register Transfer Language, Register transfer, Bus and Memory transfer, Arithmetic Micro-operations, Logic Micro-operations, Shift Micro-operations, Arithmetic Logic Shift Unit.

### UNIT-III





**Basic Computer Organization and Design:** Instruction Codes, Computer Registers; Common bus system; Computer Instructions; Instruction formats; Instruction Cycle; Fetch and Decode, Flowchart for Instruction cycle; Register reference instructions, Addressing Modes.

**CPU Design:** Specifying a CPU, design and implementation of a simple CPU (fetching instructions from memory, decoding and executing instructions, establishing required data paths).

#### UNIT-IV

**Input-Output Organization :** Input-output Interfaces, Asynchronous Data Transfer, Mode of Transfer - Programmed I/O, Interrupt I/O, Direct Memory access(DMA).

**Memory Organization:** Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory. I/O Interrupt, types of Interrupts, Priority Interrupts, Direct Memory Access(DMA).

#### Recommended Books

1. M, MorrisMano; Computer System Architectures; III Edition, Prentice Hall of India,2008
2. AndrewS.Tanenbaum , Structured Computer Organization,Printice Hall
3. WilliamStallings, Computer Organization and Architecture , Sixth Edition, Pearson
4. JohnD.Carpinelli: Computer Systems Organization & Architecture; 3<sup>rd</sup> Edition; Person Education Asia,2008
5. MalvinoB ; Digital Computer Electronics III Edition; TMHL.

## Course Content for BCA, Semesters III and IV 2024-25

### Semester: III

Course Code: <b>BCA-63T-201</b>	Course Title: <b>Data Structures and Algorithms</b>
Course Credit : 04	Hours/Week: 04

#### Course Objectives (COs):

By the end of this course, students will be able to:

- To design efficient algorithms using various algorithm designing strategies
- To analyze the problem and develop the algorithms related to these problems
- To classify the problem and apply the appropriate design strategy to develop algorithm
- To design algorithm in context of space and time complexity and apply asymptotic notation

#### Course Outcomes (COs):

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

- Understand the concepts of data structures and their importance in computing.
- Implement and use common data structures such as arrays, linked lists, stacks, queues, trees, and graphs.
- Analyze the time and space complexity of algorithms related to data structures.
- Apply data structures to efficiently solve real-world problems

### **BCA-63T-201 : Data Structures and Algorithms**

#### **UNIT – I**

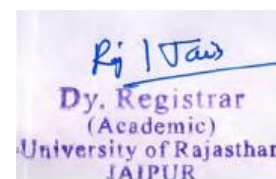
**Introduction to Algorithm Design:** Algorithm, its characteristics, efficiency of algorithms, analyzing Algorithms and problems.

**Linear Structure:** Arrays, records, stack, operation on stack, implementation of stack as an array, queue, types of queues, operations on queue, implementation of queue.

#### **UNIT – II**

**Linked Structure :** List representation, Polish notations, operations on linked list - get node and free node operation, implementing the list operation, inserting into an ordered linked list, deleting, circular linked list.

**Tree Structure :** Concept and terminology, Types of trees, Binary search tree, inserting, deleting and searching into binary search tree, tree traversals.



### UNIT – III

**Graph Structure** : Graph representation - Adjacency matrix, adjacency list, Warshall's algorithm, adjacency multilist representation. Orthogonal representation of graph . Graph traversals - BFS and DFS. Shortest path, transitive closure.

### UNIT – IV

**Searching and sorting** : Searching - sequential searching, binary searching, hashing. **Sorting** - selection sort, bubble sort, quick sort, heap sort, merge sort, and insertion sort, efficiency considerations.

#### *Recommended reference books*

1. S.Lioschutz: Data Structures, Mc Graw Hill International Edition.
2. A.V.Aho., J.E.Hopcroft, and J.D.Ullman, Data Structures and Algorithms, Pearson.
3. A. MichaelBerman: Data Structures via C++, OxfordUniversity Press.
4. SaraBaase and AllenVan Gelder: Computer Algorithms, Pearson Education Asia.

Corse Code: <b>BCA-63P-202</b>	Course Title: <b>Data Structures Lab Using C</b>
Course Credit : <b>02</b>	Hours/Week: <b>04</b>

#### **Content : Recommended exercises :**

1. Given {4,7,3,2,1,7,9,0} find the location of 7 using Linear and Binary search and also display its first occurrence.
2. Given {5,3,1,6,0,2,4} order the numbers in ascending order using Bubble Sort Algorithm
3. Perform the Insertion and Selection Sort on the input {75,8,1,16,48,3,7,0} and display the output in descending order.
4. Given {5,3,1,6,0,2,4} order the numbers in ascending order using Quick Sort Algorithm
5. Given {5,3,1,6,0,2,4} order the numbers in ascending order using Merge Sort Algorithm
6. Write a program to insert the elements {61,16,8,27} into singly linked list and delete 8,61,27 from the list. Display your list after each insertion and deletion.
7. Write a program to insert the elements {61,16,8,27} into linear queue and delete three elements from the list. Display your list after each insertion and deletion.
8. Write a program to insert the elements {61,16,8,27} into circular queue and delete 4 elements from the list. Display your list after each insertion and deletion.
9. Write a program to insert the elements {61,16,8,27} into ordered singly linked list and delete 8,61,27 from the list. Display your list after each insertion and deletion.
10. Write a program to add  $6x^3+10x^2+0x+5$  and  $4x^2+2x+1$  using linked list.
11. Write a program to push 5,9,34,17,32 into stack and pop 3 times from the stack, also display the popped numbers.

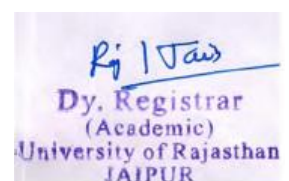
12. Write a recursive program to find GCD of 4,6,8.
13. Write a program to insert the elements {5,7,0,6,3,9} into circular queue and delete 6,9&5 from it(using linked list implementation)..
14. Write a program to create a binary tree with the elements {18,15,40,50,30,17,41} after creation insert 45 and 19 into tree and delete 15,17 and 41 from tree. Display the tree on each insertion and deletion operation
15. Write a program to create binary search tree with the elements {2,5,1,3,9,0,6} and perform inorder, preorder and post order traversal.
16. Write a program to Sort the following elements using heap sort {9,16,32,8,4,1,5,8,0}
17. Write a program to implement DFS search in a graph.
18. Write a program to implement DFS search in a graph.

Course Code: <b>BCA-63T-203</b>	Course Title: <b>Object Oriented Programming Through C++</b>
Course Credit : 04	Hours/Week: 04

### Course Objectives (COs):

Upon finishing the course, students will be able to:

- Introduction to Object-Oriented Programming: Understand the fundamental concepts of object-oriented programming (OOP) and how it differs from procedural programming.
- Java Syntax and Semantics: Learn the basic syntax and semantics of the Java programming language, including data types, operators, and control structures.
- Classes and Objects: Understand the principles of classes and objects in Java. Learn how to define and instantiate classes, and how to use objects to encapsulate data and methods.
- Inheritance and Polymorphism: Explore the concepts of inheritance and polymorphism in Java. Understand how to create class hierarchies and use polymorphic behavior to enhance code flexibility and reusability.
- Encapsulation and Abstraction: Learn about encapsulation and abstraction principles. Understand how to use access modifiers to protect data and methods, and how to define abstract classes and interfaces.
- Exception Handling: Understand the mechanisms for error and exception handling in Java. Learn how to use try-catch blocks, create custom exceptions, and manage resources with the try-with-resources statement.



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

- Understand the fundamental concepts of object-oriented programming.
- Design and implement C++ programs using classes, objects, and inheritance.
- Apply polymorphism and templates to develop reusable code.
- Utilize advanced features of C++ to develop efficient and modular programs.

## **BCA-63T-203 : Object Oriented Programming Through C++**

### **UNIT – I**

**Introduction to Object Oriented Concepts:** Evolution of OOP, OOP Paradigm, advantages of OOP, comparison between functional programming and OOP approach, characteristics of object oriented language – objects, classes, inheritance, reusability, user defined data types, polymorphism, overloading.

### **UNIT – II**

**Introduction to C++:** C++ tokens, data types, C++ operators, type conversion, variable declaration, arrays, statements, expressions, conditional statements, Jumping statements, loops, functions, pointers, structures.

**Classes and Objects:** Classes, objects, defining member functions, arrays of class objects, pointers and classes, passing objects, constructors, types of constructors, destructors, this pointer, access specifiers, friend functions, inline functions.

### **Unit – III**

**Inheritance:** Introduction, Importance of Inheritance, types of inheritance, Constructor and Destructor in derived classes., member access control.

**Polymorphism:** Functions Overloading, Operator Overloading, early binding polymorphism with pointers, Unary and Binary Operator Overloading, Overload Assignment Operator, Copy Constructor.

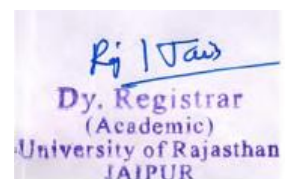
### **Unit –IV**

**Virtual Function :** Virtual Function, late binding, pure virtual functions, abstract classes, Generic Programming with Templates, Friend function, Overloaded Function Templates, Multiple Arguments function Template.

**File Management:** Handling Data files (sequential and random), Opening and closing of files, stream state member functions, Operations on File, Exception Handling.

### **Recommended Books:**

1. Deitel HM & Deitel JP; C/C++ How to program; 5<sup>th</sup>Edn; Pearson Pub.
2. Balagurusamy ; Object Oriented Programming in C++; 4<sup>th</sup> Edition TMH.
3. Venugopal, Rajkumar; Mastering C++; Tata Mcgrow Hill,.



4. KanetkarY.: LET US C++; BPB;
5. ByronGottfried; Programming with C;TMH;

Corse Code: <b>BCA-63P-204</b>	Course Title: <b>OOP Lab</b>
Course Credit : 02	Hours/Week: 04

**Course Contents : Recommended exercises**

1. Simple C++ applications for understanding references to an instant of a class
2. Handling Arrays and strings in C++
3. Inheritance applications
4. Functions overloading
5. Operators overloading
6. Use Virtual functions
7. Generic programming
8. Exception Handling
9. File operations

Corse Code: <b>BCA-63T-205</b>	Course Title: <b>Software Engineering</b>
Course Credit : 06	Hours/Week: 06

**Course Objectives (COs):**

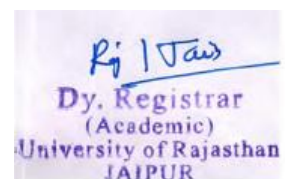
By the end of this course, students will be able to:

- Understand the principles and practices of software engineering.
- Apply software engineering processes and methodologies to develop software systems.
- Perform requirements analysis and software design.
- Implement software using appropriate programming languages and development tools.
- Apply software testing and quality assurance techniques.

**Course Outcomes (COs):**

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

- Understand the principles and practices of software engineering.
- Apply software engineering processes and methodologies to develop software systems.
- Perform requirements analysis and software design.



- Implement software using appropriate programming languages and development tools.
- Apply software testing and quality assurance techniques.

## **BCA-63T-205: Software Engineering**

### **Unit-I**

**Software Engineering Fundamentals:** Software, Problem Domain, Software Engineering Challenges, Software Processes (processes, projects & products, component), Software Requirement Analysis & Specification.

**Software Development Process Models:** Waterfall Model, Prototyping, Iterative Enhancement Model, Spiral Model. Introduction to Agile Model: Principles, Steps, Various Agile Process Models.

### **Unit-II**

**Software Project Planning:** Cost Estimation- Uncertainties in Cost Estimation, Building Cost Estimation Models, On Size Estimation, COCOMO Model.

**Project Scheduling:** Average Duration Estimation, Project Scheduling & Milestones. Quality Assurance Plans: Verification & Validation, Inspection & Reviews.

### **Unit-III**

**Design Engineering:** Design Process & Design Quality, Design Concepts (abstraction, architecture, modularity, functional independence, refinement, and design classes), The Design Model (data design elements, architectural design elements, interface design elements, component-level design elements, deployment-level design elements).

**Testing Strategies & Tactics:** A strategic approach to software testing, Strategic issues, Software testing fundamentals, Test characteristics, Test Strategies for conventional software: Unit Testing, Integration testing, Validation Testing, System testing, Black-Box testing, White Box testing.

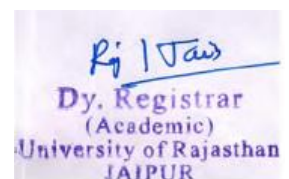
### **Unit-IV**

**Software Reliability:** Risk Management, Measures of Reliability & Availability, Software Safety.

**Maintenance and Reengineering:** Introduction to: Software Maintenance, Software Supportability, Reengineering, Reverse Engineering, Restructuring, and Forward Engineering.

#### **Reference /Text Books**

1. Pressman, Roger (2001) Software Engineering; A Practitioner's Approach, 8<sup>th</sup> ed. M Graw-Hill,2014.
2. Sommerville Ian; Software Engineering, 9<sup>th</sup>Ed.Pearson Education,2014
3. Jalote, Pankaj (7) An integrated Approach to Software Engineering 2<sup>nd</sup>Ed.
4. JamesRumbaugh. MichealBlaha, "Object oriented Modeling and Design with UML", 2<sup>nd</sup> Edition, 2007.
5. SimonBennett, SteveMcRobb and RayFarmer, " Object-Oriented Systems Analysis and Design Using UML" 4<sup>th</sup>Edition,McGraw Hill Education, 2010



*Rij / Jas*  
Dy. Registrar  
(Academic)  
University of Rajasthan  
JAIPUR



## Semester: IV

Course Code: <b>BCA-64T-211</b>	Course Title: <b>PHP Programming</b>
Course Credit : 04	Hours/Week: 04

### Course Objectives (COs):

In this course, you will learn about:

- Fundamental concepts of PHP scripting language
- The basic structure of a web application
- The request/response cycle
- Basics of MySQL database
- The relationship between the client-side and server-side scripts
- Creating functional websites and web apps in PHP
- PHP web application testing and security
- Creating a PHP web application using a CMS

### Course Outcomes (COs):

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

- Able to learn Core-PHP, Server Side Scripting Language.
- Able to design dynamic and interactive web pages, websites.
- Able to run PHP scripts on server and retrieve results.
- Able to handle databases like MySQL using PHP in web sites.

## BCA-64T-211: PHP Programming

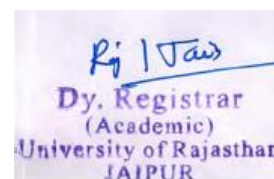
### UNIT-I

**Introduction to PHP:** Installation of PHP and MySQL, PHP configuration in IIS & Apache Web Server. Features of PHP, Writing PHP, Parsing PHP code, Embedding PHP and HTML Executing PHP and viewing in Browser.

### Unit - II

**Control Structures:** Data types, Operators, PHP variables: static and global variables, Comments in PHP, Control Structures, Condition statements, If...Else, Switch, ? operator, Loops, While, Break Statement Continue. Do...While, For, For each, Exit, Die, Return. Arrays: Numeric, Associative and Multidimensional Arrays

### UNIT-III



**Strings:** Creating and accessing String, Searching & Replacing String, Formatting String, String Related Library function, Pattern matching, Replacing text, Splitting a string with a Regular Expression

**Functions:** Defining a Function, Calling a Function, Parameter passing, Returning value from function

**Form Data Handling:** \$\_GET, \$\_POST, \$\_REQUEST Variables, Cookies handling, Session Management

#### UNIT-IV

**Exception Handling:** Understanding Exception and error, Try, catch, throw

**File Handling:** Opening and closing a file, Copying, renaming and deleting a file

**Database Handling:** Connection with MySql Database or ODBC, Performing basic database, operation (Insert, Delete, Update, Select, Truncate Alias, Order By), Setting query parameter.

#### References

1. PHP, The CompleteReference, StevenHolzner, TMH
2. Beginning PHP 5.3, MattDoyle, John Wiley & Sons
3. Core PHP Programming Leon Atkinson Pearson publishers
4. Beginning PHP 5.0 Database ChristopherScollo, Harish,Rawat, DeepakThomas,Wrox Press

Corse Code: <b>BCA-64P-212</b>	Course Title: <b>PHP Lab</b>
Course Credit : <b>02</b>	Hours/Week: <b>04</b>

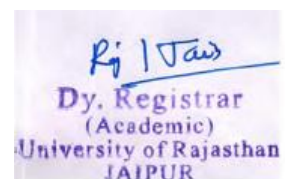
#### **Content : Recommended exercises :Exercise based on paper BCA-64T-211**

1. Installing XAMMP
2. Variables, Data Types, Constants, Operators, Programming Loops,
3. PHP Functions,
4. Arrays
5. Strings Functions
6. PHP Form Handling, Require & Include
7. PHP with MySQL

Corse Code: <b>BCA-64T-213</b>	Course Title: <b>Object Oriented Concepts Using Java Programming</b>
Course Credit : 04	Hours/Week: 04

#### **Course Objectives (COs):**

By the end of this course, students will be able to:



- Learn the Object-Oriented Programming concepts to write, compile and debug programs using Java language.
- Apply the concepts of object-oriented programming like polymorphism, inheritance, Exception Handling, and Multithreading.
- Design and develop console and GUI applications using Java Programming Language.
- Work on programming project as individual or as team member in design, development and implementation phase.

### 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 and event driven programming and also programs based on files

## BCA-64T-213: Object Oriented Concepts Using Java Programming

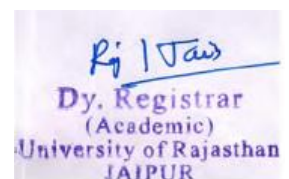
### UNIT – I

**Java Programming :** Basic concepts of object oriented programming (Objects and Classes, Data Abstraction & Encapsulation, Inheritance, Polymorphism, Dynamic binding, Message passing), Java features, JVM, Byte code interpretation, simple java program, command line argument, Data types, type casting, operators (Arithmetic, increment, decrement, relational, logical, bit wise, conditional) and expressions.

### UNIT – II

**Decision Making and Branching :** Decision making and branching (if...else, else if, switch), looping, classes, objects and methods, visibility control, constructors, wrapper classes, nesting of methods, Arrays and strings handling. **Polymorphism:** Function overriding, Operator overloading, final classes.

### UNIT – III



**Inheritance & Multithreaded Programming :** Inheritance, Types of Inheritance, Abstract class, interfaces, packages, multithreaded programming, extending thread, life cycle of thread, using thread methods, thread priority, synchronization.

**Exception Handling :** Exception-Handling fundamentals, Exception types, try, catch, throw, finally, creating exception sub classes.

#### UNIT – IV

**JSP :** Introduction to JSP, Directory Structure, Lifecycle JSP, Scripting Elements .

JAR files, Servlets Life cycle of servlet, JDBC connectivity.

#### Recommended Text Books

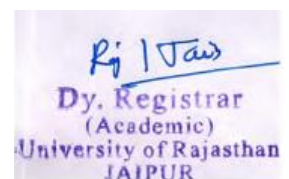
1. Mastering java 2 ”, BPB Publications. Programming with Java A Primer, E.Balagurusamy Tata McGraw Hill Companies
2. Java Programming JohnP.FlyntThomson2nd
3. The complete reference JAVA2, Herbertschildt. TMH
4. Arnold,Gosling, “ The Java Programming Professional 2000”, AddisonWesley Publication
5. C.Thomaswu, ”An introduction to oop with Java”, TMH

Corse Code: <b>BCA-64P-214</b>	Course Title: <b>Java Lab</b>
Course Credit : <b>02</b>	Hours/Week: <b>04</b>

#### Content : Recommended exercises :

1. Simple java applications for understanding references to an instant of a class
2. Handling Arrays in JAVA
3. Handling strings in JAVA
4. Implementation polymorphism
5. Package creation
6. Developing user defined packages in java
7. Use of Inheritances
8. Use of Interfaces
9. Threads, Multithreading
10. Collection handling
11. GUI/Swings applications
12. I/O Stream handling
13. Exception Handling
14. JSP
15. Servlets

Corse Code: <b>BCA-64T-215</b>	Course Title: <b>Mathematics &amp; Statistics</b>
Course Credit : <b>06</b>	Hours/Week: <b>06</b>



## Course Objectives (COs):

By the end of this course, students will be able to:

- Be able to apply problem-solving and logical skills
- Have a deeper understanding of mathematical theory
- Have a solid knowledge of elementary statistics
- Be able to communicate mathematical/logical ideas in writing
- Be competent in computer programming
- Be familiar with several subfields of mathematics (e.g, numerical analysis, topology, operations research).
- Be exposed to undergraduate research or internship opportunities

## Course Outcomes (COs):

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

- To understand the basic concepts of Mathematical reasoning, set and Relations and functions.
- To understand the basic concepts of functions and logic/proofs.
- To understand various counting techniques.
- Understand the concepts of various types of Matrices and Determinants.
- To understand the concept of Statistics: Data collection and Classification.
- To understand the concept of measurement of central tendency.
- To understand the correlation and regression analysis.

## BCA-64T-215: Mathematics & Statistics

### UNIT – I

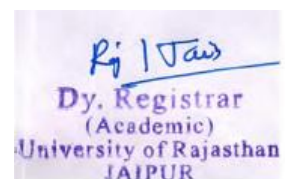
**Sets :** Definition of sets, representation of sets, type of sets, Operations on sets, Sub sets, Power set, Universal set, Complement of a set, Union and Intersection of two sets, Venn diagrams, Principles of Inclusion and Exclusion.**Relations:** Cartesian product of sets, Definition of relation, Types of relations- reflexive, symmetric, anti-symmetric, transitive , equivalence.

**Functions :** Definition, Domain & Range of a functions, one to one and onto functions, Bijective functions, composite functions, inverse of functions.

### UNIT – II

**Logic and Proofs :** Proposition, Conjunction, Disjunction, Negation, Compound proposition, De Morgan's laws, Tautology and Contradiction.

**Matrices:** Definition and Types of Matrices, Addition , Subtraction and Multiplication of Matrices, Non-commutativity of multiplication of matrices, Scalar Multiplication, Transpose of a Matrix.



**Determinant:** Determinant of a square matrix (up to 3x3 matrices), properties of determinants, minors, cofactors, expansion of determinants, application of determinants in finding the area of a triangle. Adjoint and Inverse of a matrix, Solution of system of linear equations by Cramer's Rule.

### UNIT –III

**Statistics :** Data collection methods, Data classification, Frequency Distribution, Graphical representation of frequency distribution. **Measures of Central Tendency-** Mean, Median, Mode, Measures of Dispersion- Mean Deviations, Standard Deviations, Variance

### .UNIT –IV

**Correlation Analysis :** Correlation, Types of Correlations, Methods of Studying Correlations, Measure of Karl Pearson's coefficient of correlation, Rank Correlation Coefficient.

**Regression Analysis:** Regression, Use of regression analysis,, Difference between Correlation and Regression Analysis, Regression Lines Equations, Properties of regression lines.

#### Reference Books:

1. C.L.Liu: Elements of Discrete Mathematics, Tata Mc-Graw Hill Publishing Company Ltd., 2000
2. Seymour Lipschutz; Discrete Mathematics;TMH.
3. KennethHRosen; Discrete Mathemtics& Its Applications; 6 Edition,MGH;
4. RichardJohnsonbaugh: Discrete Mathematics, Pearson Education, Asia, 2001
5. JohnTruss: Discrete Mathematics for Computer Scientists, Pearson Education, Asia, 2001.
6. Basic Mathematics, R.D.Sharma
7. B.L.Agrawal; Basic Statistics; KhannaPub.
8. S.P.Gupta; Statistical Methods; Sultan Chand & Sons
9. S.C.Gupta, V.K.Kapoor ; fundamental of statics; Sultan Chand & Sons

## Course Content for BCA, Semesters V and VI 2025-26

### Semester: V

Course Code: <b>BCA-75T-301</b>	Course Title: <b>Artificial Intelligence &amp; Machine Learning</b>
Course Credit : 04	Hours/Week: 04

### Course Objectives (COs):

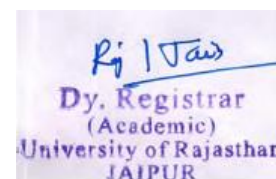
By the end of this course, students will be able to:

- Understand and explain the definition, components, and application areas of Artificial Intelligence
- Understand the concepts of Logical Reasoning and their use in knowledge representation and knowledge processing
- Describe the architecture and working of knowledge-based systems (Expert systems)
- To **Understand Machine Learning Fundamentals**, principles, and types of machine learning, including supervised, unsupervised, and reinforcement learning.
- To **Implement Machine Learning Algorithms** such as linear regression, logistic regression, decision trees, support vector machines, k-nearest neighbors, and neural networks.
- To understand and apply data preprocessing techniques, including data cleaning, normalization, and feature engineering, to prepare data for machine learning models.
- To **Evaluate Model Performance** and interpret the performance of machine learning models using appropriate metrics, such as accuracy, precision, recall, F1-score, and ROC-AUC.

### Course Outcomes (COs):

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

- Understand the fundamental concepts and techniques of artificial intelligence.
- Apply AI algorithms and methodologies to solve real-world problems.
- Design and develop AI systems using appropriate tools and frameworks.
- Evaluate and optimize AI models for performance and accuracy.
- Differentiate between supervised and unsupervised learning tasks.
- Differentiate between linear and non-linear classifiers. 3
- Describe theoretical basis of SVM



# BCA-75T-301: Artificial Intelligence & Machine Learning

## UNIT – I

**General Issues and overview of AI :** Concept of Intelligence, AI intelligent agents, Characteristics of AI, Comparison of AI, Machine Learning and Deep Learning. Defining problem as a State Space Search, Search and Control Strategies, Production systems, Problems – Water Jug problem, Block words Problem, Monkey & Banana problem, Applications of AI. Ethical considerations in AI development & deployment and MYCIN.

## Unit-II

**Searching-** Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Informed search strategies (Heuristic search) Generate-and-test, Hill climbing, Best First Search , Constraint Satisfaction ,A\*, AO\* Algorithms, Problem reduction, Game Playing-Adversarial search.

**Knowledge Representation :**Definition of Knowledge, Types of knowledge (Procedural and Declarative knowledge),Approaches to Knowledge Representation, Knowledge representation using Propositional and Predicate logic , Conversion to clause form, Resolution in Propositional logic, Resolution in Predicate logic.

## Unit-III

**Concepts :** Machine Learning, Machine Learning Foundations-Overview, Applications, Types of Machine Learning, Basic Concepts in Machine Learning – Examples of Machine Learning.

**Supervised Learning :** Introduction, Linear Models of Classification – Decision Trees, NaïveBayes Classification, Linear Regression – Logistic Regression – Bayesian Logistic Regression – Probabilistic Models Neural Network-Feed Forward Network Functions – Error Back Propagation – Regularization .

## Unit-IV

**Unsupervised Learning :** Clustering, Association rule mining, K-Means Clustering, EM (Expectation Maximization), Mixtures of Gaussians, EM algorithm in General, The Curse of Dimensionality, Dimensionality Reduction, Factor Analysis, Principal Component Analysis.

**Probabilistic Graphical Models :** Directed Graphical Models, Bayesian Networks, Exploiting Independence Properties, From Distributions to Graphs, Examples – Markov Random Fields – Inference In Graphical Models – Learning - Naïve Bayes Classifiers – Markov Models – Hidden Markov Models.

### Recommended Books:

1. ElaineRich and KevinKnight, “Artificial Intelligence”, Tata McGraw Hill, 3<sup>rd</sup> edition, 2012.
2. DanW.Patterson, “Introduction to Artificial Intelligence and Expert Systems”, Prentice Hall of India, 1<sup>st</sup> edition, 2012
3. Winston, Patrick, Henry, “Artificial Intelligence”, Pearson Education, 3<sup>rd</sup> edition, 2014





4. SubhasreeBhattacharjee, “Artificial Intelligence for Student” Shroff Publishers and Distributors Pvt.LTD., 1st Edition, 2016
5. StuartRussell, PeterNorving, “Artificial Intelligence: A Modern Approach”, Pearson Education, 3<sup>rd</sup> edition, 2010.
6. MitchellT.M., Machine Learning, McGraw Hill
7. Bishop C., Pattern Recognition and Machine Learning, Springer-Verlag
8. JoelGrus, “Data Science from Scratch-First Principles with Python”, O’Reilly, 2015
9. M. Gopal, “Applied MACHINE LEARNING”, McGraw-Hill, 2018
10. Dr. Mahaveer Kumar Sain, “Introduction to Machine Learning”, Akinik Publications-New Delhi, 2021.

Corse Code: <b>BCA-75P-302</b>	Course Title: <b>Machine Learning Lab</b>
Course Credit : <b>02</b>	Hours/Week: <b>04</b>

**Content : Recommended exercises :**

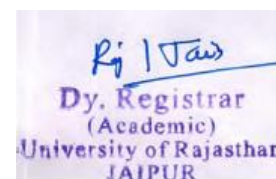
1. Install and set up Python and essential libraries like NumPy and pandas.
2. Introduce scikit-learn as a machine learning library. 3
3. Install and set up scikit-learn and other necessary tools.
4. Write a program to Load and explore the dataset of .CVS and excel files using pandas.
5. Write a program to Visualize the dataset to gain insights using Matplotlib or Seaborn by plotting scatter plots, bar charts.
6. Write a program to Handle missing data, encode categorical variables, and perform feature scaling.
7. Write a program to implement a k-Nearest Neighbours (k-NN) classifier using scikit- learn and Train the classifier on the dataset and evaluate its performance.
8. Write a program to implement a linear regression model for regression tasks and Train the model on a dataset with continuous target variables.
9. Write a program to implement a decision tree classifier using scikit-learn and visualize the decision tree and understand its splits.
10. Write a program to Implement K-Means clustering and Visualize clusters.

Corse Code: <b>BCA-75T-303</b>	Course Title: <b>Python Programming</b>
Course Credit : 04	Hours/Week: 04

**Course Objectives (COs):**

By the end of this course, students will be able to:

- To gain a solid foundation in the basics of Python programming, including syntax, data types, control structures, and functions.



- To enhance problem-solving abilities by applying Python programming techniques to solve a variety of computational problems.
- Learn to efficiently use and implement built-in data structures such as lists, tuples, dictionaries, and sets.
- To Utilize Python for data analysis and scientific computing using libraries such as NumPy, Pandas, and Matplotlib.

### **Course Outcomes (COs):**

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

- Understand the basics of programming language
- Develop, document, and debug modular Python programs.
- Apply suitable programming constructs and built-in data structures to solve a problem.
- Use and apply various data objects in Python.
- Use classes and objects in application programs and handle files.

## **BCA-75T-303: Python Programming**

### **UNIT-I**

**Python Concepts:** Origin, Comparison, Comments, Variables and Assignment, Identifiers, Basic Style Guidelines, Standard Types, Internal Types, Operators, Built-in Functions, Numbers and Strings. Sequences: Strings, Sequences, String-Operators & functions, Special Features of Strings, Memory Management, programs & examples.

**Conditionals and Loops:** if statement, else Statement, elif Statement, while Statement, for Statement, break Statement, continue Statement, pass Statement, else Statement

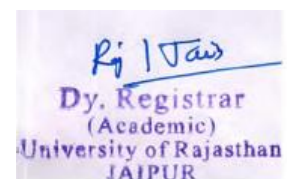
### **Unit-II**

**Object and Classes:** Classes in Python, Principles of Object Orientation, Creating Classes, Instance Methods, Class variables, Inheritance, Polymorphism, Type Identification, Python libraries(Strings, Data structures & algorithms).

**Lists and Sets:** Built-in Functions, List type built in Methods, Tuples, Tuple Operators, Special Features of Tuples, **Set:** Introduction, Accessing, Built-in Methods (Add, Update, Clear, Copy, Discard, Remove), Operations (Union, Intersection, Difference).

### **Unit-III**

**Dictionaries :** Introduction to Dictionaries, Built-in Functions, Built-in Methods, Dictionary Keys, Sorting and Looping, Nested Dictionaries.



**Files:** File Objects, File Built-in Function, File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules.

**Regular Expression:** Regular Expression: Introduction/Motivation, Special Symbols and Characters for REs, REs and Python.

## Unit-IV

**Excetiptions:** Concepts of Exceptions, Exceptions in Python, Detecting and Handling Exceptions, Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions.

**Database Interaction :** SQL Database Connection using Python, Creating and Searching Tables, Reading and storing config information on database, Programming using database connections, **Python Multithreading:** Understanding threads, Forking threads, synchronizing the threads, Programming using multithreading.

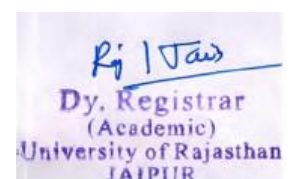
### Recommended Books:

1. R. NageswaraRao, “Core Python Programming”, Dreamtech Press, 2<sup>nd</sup> Edition, 2018
2. Dr. M. Suresh Anand, Dr. R. Jothikumar, Dr. N. Vadivelan, “Python Programming” , Notion Press, 1<sup>st</sup>Edition, 2020
3. MartinC.Brown, “The Complete Reference Python”, McGraw Hill Education, 4<sup>th</sup>Edition, 2021.
4. AshokNamdevKamthane; “Programming and Problem Solving with Python”;2<sup>nd</sup> Edn, MGH,2020
5. AllenB.Downey, “Think Python”, O’Reilly Media, 2016
6. SakisKasampalis, Quan Nguyen, Dr Gabriele Lanaro,Ingram, “Advanced Python Programming”, short title, 2019
7. DavidM.Beazley, “Python Essential Reference”, Amazon Books, 2010.
8. M. Lutz, “Programming Python, 4th Edition”, O’Reilly Media, 2010
9. DeepakThomas,Wrox Press

Corse Code: <b>BCA-75P-304</b>	Course Title: <b>Python Lab</b>
Course Credit : <b>02</b>	Hours/Week: <b>04</b>

### Content : Recommended exercises :

1. Write a program to demonstrate basic data type in python
2. Create a list and perform the following methods 1) insert() 2) remove() 3) append() 4) len() 5) pop() 6) clear()
3. Create a tuple and perform the following methods 1) Add items 2) len() 3) check for item in tuple 4)Access items
4. Create a dictionary and apply the following methods 1) Print the dictionary items 2) access items 3) use get() 4)change values 5) use len()
5. Write a program to create a menu with the following options 1. TO PERFORM ADDITITON 2. TO PERFORM SUBTRACTION 3. TO PERFORM MULTIPLICATION 4. TO PERFORM DIVISION Accepts users input and perform the operation accordingly. Use functions with arguments.
6. Write a python program to print a number is positive/negative using if-else.



7. Write a program for filter() to filter only even numbers from a given list.
8. Write a python program to print date, time for today and now
9. Write a python program to add some days to your present date and print the date added.
10. Write a program to count the numbers of characters in the string and store them in a dictionary data structure
11. Write a program to count frequency of characters in a given file.
12. Using a numpy module create an array and check the following: 1. Type of array 2. Axes of array 3. Shape of array 4. Type of elements in array
13. Write a python program to concatenate the dataframes with two different objects
14. Write a python code to read a csv file using pandas module and print the first and last five lines of a file.
15. Write a python program which accepts the radius of a circle from user and computes the area (use math module)
16. Use the following data (load it as CSV file) for this exercise. Read this file using Pandas or NumPy or using in-built matplotlib function.
  - a. Get total profit of all months and show line plot with the following Style properties  
Generated line plot must include following Style properties: –

Corse Code: <b>BCA-75T-305</b>	Course Title: <b>Data Communication &amp; Computer Networks</b>
Course Credit : 06	Hours/Week: 06

**Course Objectives (COs):**

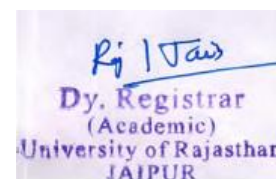
By the end of this course, students will be able to:

- Understand the basic concepts, types of networks, OSI, ans TCP/IP models with working of all the layers in detail
- Learn and understand the working of different hardware components used in networking and various communication protocols
- Learn and understand various issues involved in network security, and methods used to implement network security.

**Course Outcomes (COs):**

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

- Understand the fundamentals of data communication and networking concepts.
- Analyze and design network architectures and topologies.
- Configure and troubleshoot network devices and protocols.
- Apply network security measures to protect data transmission.



- Understand emerging trends and technologies in data communication and networks.

## **BCA-75T-305: Data Communication & Computer Networks**

### **UNIT-I**

**Introduction:** Network definition, Network topologies, Types of Network, Layered network architecture, Categories of Network, protocol, Standards and interface.

**Network Models :** client-server, peer-to-peer, OSI reference model, Architecture and functions of layers. TCP/IP protocol suite.

### **UNIT-II**

**Data Communication Fundamentals:** Analog and digital signal, Data-rate limits, Digital to digital & Digital to analog modulation. Guided and Unguided Transmission media

**Data Link Layer and Network Devices Data link layer:** framing, error detection and Corrections, flow control, Network devices: switches, routers, bridges, etc., MAC addressing and Ethernet standards.

### **UNIT-III**

**Networks Layer Functions and Protocols:** Routing, Routing algorithms, Network layer protocol of Internet- IP protocol.

**Transport Layer Functions and Protocols:** Transport services, Berkeley socket interface overview, Transport layer protocol of Internet- UDP and TCP. Overview of Application layer protocol, DNS protocol, WWW & HTTP protocols.

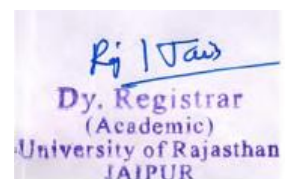
### **UNIT-IV**

**Circuit Switching :** Simple Circuit Switching, Circuit Switching Networks, Space Division switching, Time Division Multiplexing, Routing in Switching Networks, Control Signals & Channels. Packet Switching concepts and principles.

**Network Security and Wireless Networks Network security concepts:** encryption, firewalls, VPN, Wireless networks and technologies.

### **Recommended Books :**

1. Behrouz A. Forouzan, "Data Communication and Networking", 4th edition, Tata McGraw Hill.
2. A. S. Tanenbaum, "Computer Networks", Pearson Education Asia, 4th Ed..
3. William Stallings, "Data and computer communications", Pearson education Asia, 7th Ed.
4. "Computer Networking: A Top-Down Approach" by James F. Kurose and Keith W. Ross.



*Raj / Jas*  
Dy. Registrar  
(Academic)  
University of Rajasthan  
JAIPUR

## Semester: VI

Course Code: <b>BCA-76T-311</b>	Course Title: <b>.NET Framework with C#</b>
Course Credit : 04	Hours/Week: 04

### Course Objectives (COs):

By the end of this course, students will be able to:

- To introduce the fundamentals of the .NET framework.
- To learn about ASP.NET for web application development.
- To explore web forms, MVC architecture, and web services in ASP.NET.
- To understand data access techniques using ADO.NET.
- To develop and deploy ASP.NET applications.

### Course Outcomes (COs):

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

- To learn fundamentals of .net framework
- To enrich knowledge about Windows Forms, Controls and ASP.NET based applications.
- To acquire skills to create web-based applications and Reports using .net technologies.
- To enrich knowledge about ADO.NET Controls and Database based applications.

## BCA-76T-311: .NET Framework With C#

### Unit-I

**Introduction to .Net framework:** Managed Code and the CLR Intermediate Language, Metadata and JIT Compilation Automatic Memory Management

The Framework Class Library: .Net objects- ASP .NET, NET web services, Windows Forms.

**Elements :** Variable and constants data types, declaration. Operators, types precedence, Expressions Program flow, Decision statements, if then if. Then.else.select.case, Loop statements while and while, do.loop. for next for each.next

### Unit-II

**Types:** Value data types Structures, Enumerations, Reference data types, arrays.

**Windows Programming:** Creating windows forms windows controls, Button, Check box, Combo box, Label, List box Radio Button, Text box, Events, Click, close deactivate, Load, mousemove, mousedown, mouseup.



**Menus and Dialog Boxes :** Creating menus, menu items, context menu, Using dialog boxes, show dialog() method.

### Unit – III

**ADO.NET :** Architecture of ADO.NET, ADO.NET providers, Connection, Command, Data Adapter, Dataset, Connecting to Data Source, Accessing Data with Data set and Data reader, Create an ADO.NET application, Using Stored Procedures.

**ASP.NET Features:** Application of States and Structure; Change the Home Directory in IIS- Add a Virtual Directory in IIS- Set a Default Document for IIS – Change Log File Properties for IIS-Stop, Start, or Pause a Web Site.

### Unit-IV

**Creating Web Controls:** Web Controls, HTML Controls, Using Internist Control, Using Input Validation Controls, Selecting Controls for Applications, Data Controls and Adding web controls to a page.

**Creating Web Forms:** Server Controls, Types of Server Controls, Adding ASP.NET Code to a page.

**Web Services and WCF :** Web Services protocol and standards – WSDL Documents-Visual Studio.NET Architecture of WCF, WCF Client

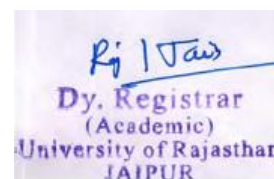
#### Recommended Books :

1. Mathew Mac Donald: Beginning ASP.NET 4.0 in C# 2010, 3<sup>rd</sup> Edition, A Pres.
2. BillEvjenScottHanselman, DevinRader: Professional ASP.NET4, 2010, Willey.
3. GeorgeShepherd: Microsoft ASP.NET Step by step, 2010 Microsoft Press.
4. ImarSpaanjaars: Beginning ASP.NET 4: in C# and VB (Wrox Programming to Programmer) , 2010 Wiely Publishing.
5. StevenHolzner; ASP.NET 4.0 (Cover C# & VB ) Black Book; Dreamtech Press.
6. StevenHolzner; .NET Programming Black Book; Dreamtech Press.

Corse Code: <b>BCA-76P-312</b>	Course Title: <b>.NET with C# Lab</b>
Course Credit : <b>02</b>	Hours/Week: <b>04</b>

#### Content : Recommended exercises :

- 1.Simple application of window programming
- 2.Use of control structures in array handling
- 3.Implementation of basic data structures
- 4.Functions overloading
- 5.Working with GUI controls
- 6.Handling of multiple classes using interfaces
- 7.Event handling with controls
- 8.Working with Data Controls





9. Dynamic data binding
10. Use of Validation controls
11. Creating Forms & Dialog boxes
12. Working with Web Controls
13. Creating & Implementation User controls
14. Create Web sites
15. Session Management
16. Exception handling using Ajax toolkit
17. Web Services and WCF.

Course Code: <b>BCA-76T-313</b>	Course Title: <b>Introduction to Data Science</b>
Course Credit : 04	Hours/Week: 04

### Course Objectives (COs):

By the end of this course, students will be able to:

- To introduce the basics of data science and its applications.
- To learn about data manipulation, visualization, and analysis using R.
- To explore statistical modeling, machine learning, and data mining techniques.
- To implement data science projects using R programming language.
- To study real-world applications and case studies of data science.

### Course Outcomes (COs):

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

- Understand the fundamental concepts and principles of data science.
- Apply data manipulation techniques using appropriate tools and libraries.
- Analyse and visualize data to extract insights and make data-driven decisions.
- Apply statistical methods and machine learning algorithms to analyse data.
- Communicate data findings effectively through data visualization and storytelling.

## **BCA-76T-313: Introduction to Data Science**

### **Unit-I**

**Introduction to Data Science** : Concept of Data Science, Need for Data Science, Components of Data Science, Big data, Facets of data: Structured data, Unstructured data, Machine-generated data, Graph-based or network data, Audio, image and video, Streaming data, The need for Business Analytics, Data Science Life Cycle, Applications of data science.



## Unit-II

**Data Science Process** : Overview of data science process, setting the research goal, Retrieving data, Cleansing, integrating and transforming data, Exploratory data analysis. Data Modeling, Presentation and automation

**Data Analytics**: Types of Analytics, Data Analytics Lifecycle: Overview - Discovery - Data Preparation - Model Planning - Model Building, Regression analysis, Classification techniques, Clustering, Association rules analysis.

## Unit-III

**Statistics** : Basic terminologies, Population, Sample, Parameter, Estimate, Estimator, Sampling distribution, Standard Error, Properties of Good Estimator, Measures of Central tendency , Measures of Spread, Probability, Normal Distribution, Binary Distribution, Hypothesis Testing ,Chi-Square Test.

## Unit-IV

**Data Science Tools and Algorithms** : Basic Data Science languages- R, Python, Knowledge of Excel, SQL Database, Introduction to Weka, Regression Algorithms - Linear Regression, Logistic Regression, K-Nearest Neighbors Algorithm, K-means algorithm.

### Recommended Books :

1. SamuelBurns, "Fundamentals of Data Science: Take the first Step to Become a Data Scientist" , Amazon KDP Printing and Publishing, First Edition, 2019
2. DavyCielen, ArnoD.B.Meysman, MohamedAli, "Introducing Data Science", Manning Publications, 2016
3. CathyO'Neil and RachelSchutt, "Doing Data Science, Straight Talk From The Frontline", O'Reilly. 2014.

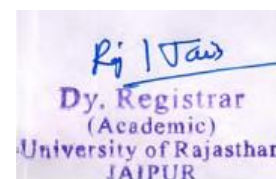
Corse Code: <b>BCA-76P-314</b>	Course Title: <b>Data Science Lab</b>
Course Credit : 02	Hours/Week: 04

**Content : Recommended exercises** :Lab Exercise based on Theory Paper BCA-75T-303.

**R Programming**: Fundamentals, Properties & Characteristics, Data Types,Operators,Control & Looping Structures, Array & String handling, Functions, Vector & Matrices processing, Factors, Data Frames, Packages, Data Reshaping, Data and File management, Charts and Graphs.

**Data science with R/Python** : Overviews, data visualisation using graphics in R, GGplot 2, File format of graphics output, introduction to hypotheses, types of hypothesis, data sampling, confidence and significance level, hypothesis tests, parametric test, non-parametric test,

**Regression Algorithms in R/Python** : How Regression Algorithm Work, Linear Regression, Logistic Regression, K-Nearest Neighbors Algorithm, K-means algorithm.



Course Code: <b>BCA-76T-315</b>	Course Title: <b>Cloud Computing</b>
Course Credit : 06	Hours/Week: 06

**Course Objectives (COs):**

By the end of this course, students will be able to:

- To understand the principles and paradigm of Cloud Computing
- Ability to design and deploy Cloud Infrastructure
- Understand cloud security issues and solutions
- Ability to understand role of Virtualization Technologies
- Design & develop backup strategies for cloud data based on features

**Course Outcomes (COs):**

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

- Understand the key dimensions of the challenges and benefits of Cloud Computing.
- Describe the principles of Parallel and Distributed Computing and evolution of cloud computing from existing technologies
- Implement different types of Virtualization technologies and Service Oriented Architecture systems.
- Choose among various cloud technologies for implementing applications.
- Install and use current cloud technologies.

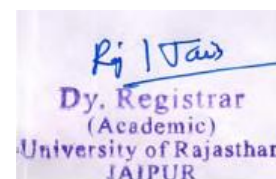
**BCA-76T-315: Cloud Computing**

**Unit-I**

**Introduction of Cloud Computing:** Definition, Historical Developments, Enabling Technology, Vision, Essential Characteristics of Cloud Computing , Components of Cloud Computing. Challenges and Approaches of Migration into Cloud, Cloud Applications: – Health care, CRM and ERP, Social Networking, Media Applications and Multiplayer Online Gaming. Benefits: For the Market, Enterprise, End user and Individuals.

**Unit-II**

**Cloud Computing Architecture :** Introduction, Cloud Reference Model, Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service, Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economics of the Cloud, Open Challenges, Cloud Interoperability and Standards, Scalability and Fault Tolerance. Parallel and distributed Computing-MapReduce, High level Language for Cloud, Service Oriented Computing.



### Unit-III

**Virtualization:** Introduction, Characteristics of Virtualized Environment, Taxonomy of Virtualization Techniques, Virtualization and Cloud computing, Virtualization: of CPU, Memory, I/O Devices, Server , Desktop, Network, and data-center. Pros and Cons of Virtualization, Technology Examples-VMware and Microsoft Hyper-V,KVM, Xen.

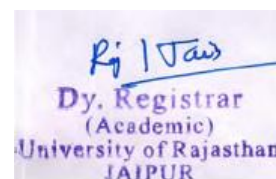
**Introduction of Cloud security services,** Design Principles, Policy Implementation, Cloud Computing Security Challenges, Cloud Computing Security Architecture. Cloud Security technologies to secure the data in Private and Public. Security Concerns. Risk Mitigation, Understanding and Identification of Threats in Cloud, SLA-Service Level Agreements.

### Unit-IV

**Cloud Platforms in Industry:** Amazon Web Services- Compute Services, Storage Services, Communication Services and Additional Services. Google AppEngine-Architecture and Core Concepts, Application Life-Cycle, cost model..

#### Recommended Books:

1. Cloud Computing ,Principle and Paradigms, Edited By RajkumarBuyya, JemesBroberg, A.Goscinski, Pub.- Wiley-2016
2. KumarSaurabh, “Cloud Computing” , Wiley Pub 2016
3. "Cloud Computing: Concepts, Technology & Architecture" by ThomasErl, RicardoPuttini, and ZaighamMahmood
4. "Cloud Computing: A Practical Approach" by AnthonyT.Velte, Toby J.Velte, and RobertElsenpeter
5. Mastering Cloud Computing by RajkumarBuyya, ChristianVecchiola, S.ThamaraiSelvi from TMH 2013.
6. Distributed and Cloud Computing, KaiHawang , GeoffreyC.Fox, JackJ.DongarraPub: Elsevier, 2013
7. Krutz , Vines, “Cloud Security “ , Wiley Pub,2010
8. Velte, “Cloud Computing- A Practical Approach” ,TMH Pub,2009
9. KatarinaStanoevska-Slabeva, ThomasWozniak, SantiRistol, “Grid and Cloud Computing – A Business Perspective on Technology and Applications”, Springer,2010



## List of Multi disciplinary Elective Courses(MEC) in Bachelor of Computer Application

S.No.	Course Code	Course Title	Theory/ Practical	Teaching Hours/ Week	Credits	To be Opted in the Semester
1	BCAMEC1	C Programming Concepts	Theory	4	4	III
2	BCAMEC2	Web Application Development	Theory	4	4	III
3	BCAMEC3	Fundamentals of Computer	Theory	4	4	III
4	BCAMEC4	Office Management Tools	Theory	4	4	III
5	BCAMEC5	E-Commerce Technologies	Theory	4	4	IV
6	BCAMEC6	Programming in Python	Theory	4	4	IV
7	BCAMEC7	Digital Marketing	Theory	4	4	IV
8	BCAMEC8	Computer Accounting	Theory	4	4	IV
9	BCAMEC9	Multimedia and Animation	Theory	4	4	V
10	BCAMEC10	Introduction to Cyber Security	Theory	4	4	V
11	BCAMEC11	Open Source with PHP	Theory	4	4	V
12	BCAMEC12	Graphic Design	Theory	4	4	V

*Rj / Jau*  
Dy. Registrar  
(Academic)  
University of Rajasthan  
JAIPUR