

**Faculty of IT & Computer Application**

**Scheme of  
BCA (Bachelor of Computer Applications)-  
Cloud Computing (-In Association with AWS)  
Academic Programme- July 2021-22**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Semester wise credit** | | | | | | **Credits** | **Minimum Credit for degree** |
| **I** | **II** | **III** | **IV** | **V** | **VI** | **Total** |
| **22** | **22** | **24** | **24** | **24** | **16** | **132** | **132** |

**Total Credits for the Batch 2021-24 = 132 Credits**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Semester-1 | | | | | | |
| Course Code | Course Name | L  (Hr.) | T (Hr.) | P(Hr.) | Credits | Type |
| BCA201 | Operating System | 3 | 0 | 0 | 3 | CORE |
| BCA202 | Programming in C | 3 | 1 | 0 | 4 | CORE |
| BCA204 | Computational Thinking and Problem Solving | 2 | 1 | 0 | 3 | F |
| AWS001A | Cloud Computing Fundamentals | 3 | 0 |  | 3 | CORE |
| BCA205 | Computer Programming Lab | 0 | 0 | 2 | 1 | CORE |
| BCA206 | Computer Systems Organization lab | 0 | 0 | 2 | 1 | CORE |
| BCA207 | Computational Thinking and Problem Solving Lab | 0 | 0 | 2 | 1 | F |
| BCA208 | Seminar | 0 | 0 | 2 | 1 | CORE |
|  | Communication Skills | 2 | 0 | 2 | 3 | ID |
|  | Culture Education-I | 2 | 0 | 0 | 2 | ID |
|  | Total | 15 | 2 | 10 | 22 |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Semester – II** | | | | | | |
| **Course Code** | **Course Name** | **L(Hr.)** | **T (Hr.)** | **P(Hr.)** | **Credits** | **Type** |
| BCA209 | Database Management System | 3 | 1 | 0 | 4 | CORE |
| BCA210 | Object Oriented Programming using C++ | 3 | 1 | 0 | 4 | F |
| AWS 002A | Introduction to Cloud Architecture | 3 | 0 | 0 | 3 | CORE |
| BCA212 | Database Management System Lab | 0 | 0 | 2 | 1 | CORE |
| BCA213 | Project Lab using C++ | 0 | 0 | 2 | 1 | F |
|  | Environmental Studies | 2 | 0 | 0 | 2 | ID |
|  | Culture Education-II | 2 | 0 | 0 | 2 | F |
|  | Professional Skills | 2 | 0 | 0 | 2 | F |
|  | Open Elective-I | 3 | 0 | 0 | 3 | GE |
|  |  |  |  |  |  |  |
|  | **Total** | 17 | 2 | 6 | 22 |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Semester – III** | | | | | | |
| **Course Code** | **Course Name** | **L(Hr.)** | **T (Hr.)** | **P(Hr.)** | **Credits** | **Type** |
| BCA216 | Data Structures and Algorithms | 3 | 1 | 0 | 4 | F |
| BCA217 | Programming in JAVA | 3 | 1 | 0 | 4 | CORE |
| BCA218 | Statistical and Numerical Methods | 3 | 0 | 0 | 3 | ID |
| AWS 003A | Cloud Security | 4 | 0 | 0 | 4 | CORE |
| BCA219 | Project Programming Lab in PHP | 0 | 0 | 2 | 1 | CORE |
| BCA220 | Data Structures Lab | 0 | 0 | 2 | 1 | F |
| BCA221 | Project Lab using JAVA | 0 | 0 | 2 | 1 | CORE |
|  | Life Skills -I (Personality Development) | 1 | 0 | 0 | 1 | F |
|  | Value Education-I | 1 | 0 | 0 | 1 | F |
|  | Life Skills -I Lab | 0 | 0 | 2 | 1 | F |
|  | Open Elective-II | 3 | 0 | 0 | 3 | GE |
|  |  |  |  |  |  |  |
|  | **Total** | **18** | **2** | **8** | **24** |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Semester – IV** | | | | | | |
| **Course Code** | **Course Name** | **L(Hr.)** | **T (Hr.)** | **P(Hr.)** | **Credits** | **Type** |
| BCA222 | Advance Java | 3 | 0 | 0 | 3 | CORE |
| BCA223 | Programming in Python | 3 | 0 | 0 | 3 | CORE |
| BCA224 | Software Engineering | 3 | 0 | 0 | 3 | F |
| BCA225 | Web Technologies | 3 | 0 | 0 | 3 | F |
| AWS 004A | Elective I/ AWS Academy Cloud Foundation Course | 3 | 0 | 0 | 3 | CORE |
| BCA226 | Project Lab (Web Technologies) | 0 | 0 | 2 | 1 | F |
| BCA227 | Project Lab in Advance Java | 0 | 0 | 2 | 1 | CORE |
| BCA228 | Project Lab in Python | 0 | 0 | 2 | 1 | CORE |
|  | Life Skills - II (Aptitude) | 1 | 0 | 0 | 1 | F |
|  | Value Education and Ethics-II | 1 | 0 | 0 | 1 | F |
|  | Life Skills - II Lab | 0 | 0 | 2 | 1 | F |
|  | Open Elective-III | 3 | 0 | 0 | 3 | GE |
|  | **Total** | 20 | 0 | 8 | 24 |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Semester – V** | | | | | | |
| **Course Code** | **Course Name** | **L(Hr.)** | **T (Hr.)** | **P(Hr.)** | **Credits** | **Type** |
| BCA229 | Computer Networks | 3 | 0 | 0 | 3 | CORE |
| BCA230 | Cryptography and Cyber Security | 3 | 0 | 0 | 3 | CORE |
| BCA231 | Mobile Application Development | 2 | 0 | 0 | 2 | F |
| BCA232 | Mobile Application Development Lab | 0 | 0 | 2 | 1 | F |
| BCA233 | Cryptography Lab | 0 | 0 | 2 | 1 | CORE |
| BCA234 | UNIX and Shell Programming Lab | 0 | 0 | 2 | 1 | CORE |
| BCA235 | Project | 0 | 0 | 2 | 1 | CORE |
| AWS 005A | AWS Academy Cloud Developing | 3 | 0 | 0 | 3 | CORE |
|  | Elective – I (202/300) | 3 | 0 | 0 | 3 | CORE |
|  | Elective – I (202/300) | 3 | 0 | 0 | 3 | CORE |
|  | Open Elective-IV | 3 | 0 | 0 | 3 | GE |
|  | **Total** | **20** | **0** | **8** | **24** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Semester – VI** | | | |
| **Course Code** | **Course Name** | **Credits** | **Type** |
| **BCA242** | **Industrial Training/Internship/Dissertation Project Presentation** | **16** | **CORE** |

**List of courses for Electives**

|  |  |  |  |
| --- | --- | --- | --- |
| **Electives-(202)** | |  | **Electives-(300)** |
| **Course Code** | **Course Name** | **Course Code** | **Course Name** |
| BCA237 | Programming in ASP.Net | BCA250 | Data warehousing and Data Mining |
| BCA238 | Programming in R | BCA251 | Data Science and Analytics |
| BCA239 | Programming in C# | BCA252 | Big Data Analytics |
| BCA240 | Professional Java | BCA253 | Cloud Computing – |
| BCA241 | Programming in ASP.Net Lab | BCA254 | Introduction to Salesforce |
| BCA242 | Programming in R Lab | BCA255 | Introduction to AI & Machine Learning |
| BCA243 | Programming in C# Lab | BCA256 | Design & Analysis of Algorithms |
| BCA244 | Professional Java Lab | BCA257 | Compiler Design |
| BCA245 | MEAN Fullstack Development | BCA258 | Theory of Computation |
| BCA246 | Introduction to Linux System Administration | BCA259 | Robotics Process Automation |
| BCA247 | Linux Server Administration and Automation |  | Mathematical Foundation |
| BCA248 | Software Development Using OpenShift Architecture |  | Statistical and Numerical Methods |
| BCA249 | Network Virtualization Using OpenStack |  |  |

**Semester – I**

**Course Name: Operating System**

**Course Code: BCA201**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 3 |

**Course Objectives::**

1. To understand the OS role in the overall computer system
2. To study the operations performed by OS as a resource manager and the scheduling policies of OS
3. To understand the different memory management techniques
4. To understand process concurrency and synchronization
5. To understand the concepts of input/output, storage and file management and to study different OS and compare their features.

**Syllabus**

**Unit – I**

**Introduction**-Operating system objectives, User view, System view, Operating system definition ,Types, Functions, Computer System Organization, Computer System Architecture, OS Structure, Operating System services, User and OS Interface, System Programs, Operating System Design and Implementation, OS Structure.

**Unit – II**

**Process and CPU Scheduling** – Process concepts, The Process, Process State, Process Control Block, Threads, Process Scheduling, Scheduling Criteria, Scheduling algorithms Scheduling Queues, Schedulers, Context Switch, Operations on Processes, System calls, Process Synchronization, The Critical Section Problem, Peterson’s solution, Synchronization Hardware, Semaphores.

**Unit – III**

**Deadlocks-** Deadlock characterization, Methods of handling deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

**Unit- IV**

**Memory Management** – Memory Management Strategies- Overlays and Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of Page Table, Virtual Memory Management- Segmentation, Paging,  Demand Paging, Page Replacement, Page Replacement Algorithms, Allocation of Frames, Thrashing.

**Unit - V**

**File System and Security** - File-System Structure, File-System Implementation , Directory Implementation, Allocation Methods, Free-Space Management, Efficiency and Performance, Recovery, NFS, Organization of I/O Function, I/O Buffering, Disk Organization, Disk Scheduling, RAID, Operating Systems view of file system, Disk space management. System Security- The Security Problem, Program Threats, Policies, System and Network Threats, User Authentication, Security Models.

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Differentiate between different types of Operating Systems and their working, functions and services

CO2: Define, restate, discuss, and explain the policies for scheduling and deadlocks

CO3: Apply techniques and algorithms for memory allocation, management, process, synchronization, system calls, and file systems

CO4: Design and construct the following OS components: System calls, Schedulers, Memory management systems, Virtual Memory and Paging systems

CO5:Measure, evaluate, and compare OS components through algorithms of disk Management and security and protection.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | H |  |  |  |  |  |  |  |  |  |  |  |
| CO2 | H |  |  | L |  |  |  |  |  |  |  |  |
| CO3 | H |  |  | L |  |  |  |  |  |  | M |  |
| CO4 | H |  | H |  |  |  |  |  |  |  |  |  |
| CO5 | H | H |  |  | M |  |  |  |  |  |  |  |

H = Highly Relevant ; M = Medium Relevant; L = Low Relevant

##### **Text Books**

1. A Silberschatz, P.B. Galvin, G. Gagne, Operating Systems Concepts, 8th edition, John Wiley Publications ,2008.
2. A.S. Tanenbaum, Modern Operating Systems, 3rd edition, Pearson Education ,2007.

##### **Reference Books**

1. G. Nutt, Operating Systems: A Modern Perspective, 2nd edition Pearson Education ,1997.
2. W. Stallings, Operating Systems, Internals & Design Principles, 5th edition, Prentice Hall of India, 2008.
3. “Operating Systems: Internals and Design Principles” by William Stallings

**Semester – I**

**Course Name: Programming in C**

**Course Code: BCA202**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 1 | 0 | 4 |

**Course Objectives:**

1. To provide exposure to problem solving through programming.
2. To enhance the ability to solve the problem through various constructs of C such as decision control and looping statements.
3. To demonstrate the concept of Accessing Array Elements, Storing Values in Arrays and length of an Array.
4. To evaluate the concept of data types and variable.
5. To explore major concepts of computer science and the process of computer programming, including programming, procedural and data abstraction and program modularity.

**Syllabus**

**Unit I**

**Introductory Concepts:** Introduction to computers**,** Computer characteristics, Modes of operation, Types of programming languages, Introduction to C, Some simple C programs, Desirable program Characteristics.

**C Fundamentals:** The C character set, Identifiers and keywords, Data Types, Constants, Variable and Arrays, Declarations, Expression, Statements, Symbolic Constants.

**Operators and Expressions:** Arithmetic operator, Unary operators, Relational and Logical operators, Assignment operators, Conditional operator, Library Functions.

**Unit II**

**Data Input and output:** Preliminaries, single character input-ge tchar function, single character output-put char function, Entering input data-The scanf function, Writing output data-The printf function, The gets and puts functions, Interactive(Conversational) programming.

**Preparing and running a complete C program:** Planning a C program, writing a C program, Entering the program into the computer, Compiling and executing the program, Error diagnostics, and Debugging techniques.

**Control Statements:** Preliminaries, Branching: The If-else Statement, Looping: The while statement, The Do while Statement, The For Statement, nested control structures, The switch statement, The break statement, The continue statement, The Comma Operator, The Goto Statement.

**Unit III**

**Functions:** A brief overview, defining a function, accessing a function, function prototype, passing arguments to a function, recursion.

**Program Structure:**  Storage classes, automatic variables, external(global) variables, static variables, multi file programs, library functions.

**Unit IV**

**Array:** Defining an array, processing an array, passing arrays to functions, multidimensional arrays, arrays and strings.

**Pointers:** Fundamentals of pointers, pointer declaration, passing pointers to functions, pointer and one dimensional arrays, Dynamic memory allocation, operations on pointers, pointers and multidimensional arrays, arrays of pointers, Passing functions to other functions.

**Unit V**

**Structures and Unions:**  Defining a structure, processing a structure, user defined data types(typedef), structures and pointers, passing structures to functions, self referential structures, Unions.

**Data files:**  Opening and closing a data file, creating a data file, processing a data file, unformatted data files.

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Learn the concept of programming language constructs and logic building for problem formulation and solution

CO2: Classify and compare the different input output methods, and different control structures available in C

CO3: Formulate and apply programs based on functions and storage classes

CO4: Develop conceptual understanding of use of arrays and pointers for constructing effective programs.

CO5: Distinguish between structure and union, and employ various techniques to store data in a file..

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | H | H | H |  |  |  |  |  |  |  |  |  |
| CO2 | H | H | H | L |  |  |  |  |  |  |  |  |
| CO3 | H | H | H |  |  |  |  |  |  |  |  |  |
| CO4 | H | H | H |  |  |  |  |  |  |  |  |  |
| CO5 | H | H | H |  |  |  |  |  |  |  |  |  |

H = Highly Relevant ; M = Medium Relevant; L = Low Relevant

**Text Books**

1. Reema Theraja , ''Programming in C", Second Edition , Oxford University Press, 2016
2. Yeshwant kanetkar, Let us C, Fifteenth Edition, BPB Publication 2016.

**Reference Books**

1. Reema Thareja, “Introduction to C Programming”, First Edition, Oxford University press, 2012.
2. Byron S Gottfriend, “Programming with C”, Second Edition, Mc Graw Hill, 2007.
3. Herbert Schildt, Turbo C**:** The Complete Reference, Fourth edition, Mc Graw-Hill, 2017.

**Semester – I**

**Course Name: Computational Thinking and Problem Solving**

**Course Code: BCA204**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 2 | 1 | 0 | 3 |

**Course Objectives:**

1. To provide exposure to problem solving through programming.
2. To enhance the ability to solve the problem through various concepts searching and sorting/
3. To demonstrate the concept of problem solving using various methods.
4. To evaluate the concept of Abstraction, greedy methods, divide and conquer method.

Unit 1

Basics Introduction, Information and data, Number Systems-Binary, Hexadecimal, Octal, Conversion, BCD, Data encoding. Boolean Algebra, Simplification of Boolean expression.

Unit 2

Problem Solving Problem definition, Problem decomposition, Abstraction, Greedy Method, Divide and Conquer.

Unit-3

Algorithmic Thinking Algorithm and Flowcharting, Name binding, Selection, Repetition.

Unit 4

Data organization: List and Arrays, Modularization, Problem Solving: Factoring and Recursion Techniques,

Unit 5

Searching and Sorting Techniques, Text processing and Pattern matching.

**Course Outcomes (COs):-**

**On successful completion of this course, the learner will be able to:**

CO1: State the concept of data, information and number system.

CO2: Analyze the problem by defining and using concepts of greedy method and divide and conquer.

CO3: Illustrate the basic structure and operation of flowchart, selection and repetition.

CO4: Develop an understanding of concepts of Array, modularization, and concepts of recursion.

CO5: Examine the searching and sorting concepts.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | H | H | M |  |  |  |  |  |  |  |  |  |
| CO2 | H | H | M |  |  |  |  |  |  |  |  |  |
| CO3 | H | H | M |  |  |  |  |  |  |  |  |  |
| CO4 | H | H | M |  |  |  |  |  |  |  |  |  |
| CO5 | H | H | M |  |  |  |  |  |  |  |  |  |

H = Highly Relevant ; M = Medium Relevant; L = Low Relevant

**Reference Books:-**

1. David Riley and Kenny Hunt , Computational thinking for modern solver, Chapman & Hall/CRC, 2014
2. R.G. Dromey , “How to solve it by Computer”, PHI, 2008

**Semester – I**

**Course Name: Cloud Computing Fundamental**

**Course Code: AWS001A**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 3 |

**Course Objectives**:-

**CO1** This course helps students to build an understanding of the fundamental concepts of cloud computing.

**CO2** Students will be able to understand theoretical and practical concepts of traditional cloud and AWS cloud services.

**Unit1- Cloud Concepts Overview**: Introduction to cloud computing, advantages of cloud, Introduction to AWS, moving to AWS Cloud.

**AWS Global Infrastructure**: AWS Global Infrastructure, AWS services and AWS categories. AWS management console.

**Unit 2- Networking in AWS**: Introduction, networking basics, Amazon VPC, VPC networking.

**Unit 3- AWS Compute Services**: Introduction, Computer Services Overview, Amazon EC2, Amazon Lambda and Amazon Beasnstalk.

**Unit 4- AWS EBS**: Introduction, Amazon Elastic Block Store Console, Working with EBS.

**Unit 5- AWS Cloud Architecture**: Introduction, AWS Well-Architected Framework Design Principles

**Automatic Scaling and Monitoring**: Introduction, Amazon Elastic Load Balancing, Amazon CloudWatch, Amazon EC2 Auto Scaling

|  |
| --- |
| **Course Outcomes (CO)** |
| 1. Students are able to understand basic cloud concepts and AWS Global Infrastructure. |
| 1. Students are able to understand the basic concepts of networking in AWS. |
| 1. Students are able to understand the basic concepts of Computing Services in AWS. |
| 1. Students are able to understand the basic concepts of AWS EBS. |
| 1. Students are able to understand the basic concepts of AWS Cloud Architecture and Automatic Scaling and Monitoring. |

**Semester – I**

**Course Name: Computer Programming Lab**

**Course Code: BCA205**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objective:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

* Operators- Arithmetic, Relational, Ternary, Logical, Bitwise
* Control Statements-if, if-else, nested if, if-else if, switch, goto
* Looping Control-while, for, do-while
* Arrays-one-dimensional- creating, displaying merging, searching, sorting, reversing
* Arrays-Two-dimensional- creating, displaying, Operations on 2D arrays
* Strings-String functions, manipulation of strings, multi strings
* Pointers – Pointer arithmetic, Array of pointers, pointer to array
* Functions – passing arguments, returning values, recursive functions, pointers as arguments
* Structures-Initializing, members as array, variables as array, passing structures to functions, pointers to structures
* Union-Enum types, preprocessors-macros, macro with arguments, nested macro, file inclusion, command line arguments
* File Handling

**Semester – I**

**Course Name: Computer System Organization Lab**

**Course Code: BCA206**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objective:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**Experiment 1 .**Ripple Carry Adder

**Experiment 2.** Carry-Look-Ahead Adder

**Experiment 3.** Wallace Tree Adder

**Experiment 4.** Synthesis of Flip Flops

**Experiment 5.** Registers and Counters

**Experiment 6.** Combinational Multipliers

**Experiment 7.** Booth's Multiplier

**Experiment 8 .** Arithmetic Logic Unit

**Experiment 9.** Memory Design

**Experiment 10.** Associative cache Design

**Experiment 11.** Direct Mapped cache Design

**Experiment 12.** CPU Design

**Experiment 13.** Karnaugh Map

**Experiment 14.** Quine - Mc Clusky Algorithm

**Semester – I**

**Course Name: Computational Thinking and Problem Solving Lab**

**Course Code: BCA207**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objectives**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their knowledge and concepts after implementation of all the given experiments.

Unit 1 Excel

**Unit-2** Excel

**Unit3**-Flowgarithm

**Unit-4**-Scratch

**Semester – I**

**Course Name: Seminar**

**Course Code: BCA208**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objective:**

The students will present a seminar on a technical topic in front of class and faculty. The evaluation will be done on individual performance followed by submission of a written report on Seminar Topic.

**Semester – I**

**Course Name: Communication Skills**

**Course code: BCA105A**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 2 | 0 | 2 | 3 |

**Course Objectives:**

1. To enhance English language competence in reading, writing, listening and speaking.
2. Switch the approach from teacher-centered to student-centered one.
3. Minimize the Grammar Translation Method of ELT while trying to replace it with Direct Method.
4. Introduce Communicative Method of ELT and focusing the teaching pedagogy on the student-centered learning rather than on the teacher-centered learning.
5. To link communication skills with the organizational behavior.
6. To inculcate skills that is very much required for employability and adjusts in the professional Environment.

**Syllabus: Theory**

|  |  |
| --- | --- |
| **UNIT 1** | **Basics of Organizational Communication:**  Communication: Meaning, Elements, Process, Types, Flows of Communication and Barriers to communication, basics of professional communication and professional ethics including Time-management, Respect for deadlines and corporate culture |
| **UNIT 2** | **Basic Writing Skills:** Parts of Speech, Elements of Sentences, Sentence types based on meaning and structure, Tenses, Voice, Narration |
| **UNIT 3** | **Composition:**, Basics of Letter Writing, Email Writing, Précis Writing, Essay Writing, |
| **UNIT 4** | **Vocabulary Building:** Word Formation from one word form to another, Origin of Words, Affixes, Synonyms, Antonyms |
| **UNIT 5** | **Professional and Technical Communication :** Basics of Drafting a CV/Resume, Basics of Telephonic Interview and Online Interview, Basics of PPT presentation |

**Syllabus: Lab**

|  |  |
| --- | --- |
| **UNIT 1** | **Basics of Organizational Communication:** Role Plays and presentations related to different corporate related matters- How to greet, how to deny politely, how to handle different types of problems related to the types of communication, how to avoid grapevine and use it in a positive manner, how to keep positive mindset during work pressure, Activities to teach Time-management, Following Deadlines etc |
| **UNIT 2** | **Write Dialogue from the different contexts of corporate culture:**  Employee and Employer, Customer and Service Provider, Customer and Product Review, How to react on Day to day corporate interactions- Memo, Notice, Email, Circular etc |
| **UNIT 3** | **Composition:**, Letter Writing, Email Writing, Précis Writing, Essay Writing,  Practice sessions by using Ms Word- Following the process of Drafting-Redrafting, Proof Reading, Editing etc |
| **UNIT 4** | **Vocabulary Building:** Word Formation from one word form to another, Origin of Words, Affixes, Synonyms, Antonyms- Using video clips and comprehension passages to find out the difference between words, similarity between words, origin of words, neologism concepts etc |
| **UNIT 5** | **Professional and Technical Communication :** Drafting a CV/Resume, Practice Sessions on Telephonic Interview and Online Interview, Presenting projects, proposals etc through PPT Making, |

**Suggested Reading:**

1. Practical English Usage. Michael Swan. OUP. 1995
2. Remedial English Grammar. F.T. Wood. Macmillan. 2007
3. Raymond V. Lesikar and Marie E. Flatley. Basic Business Communication, Tata McGraw Hill Pub. Co. New Delhi. 2005. Tenth Edition.
4. On Writing Well. William Zinsser. Harper Resource Book. 2001
5. Study Writing. Liz Hamp-Lyons and Ben Heasly. Cambridge University Press. 2006.
6. Communication Skills. Sanjay Kumar and PushpLata. Oxford University Press. 2011.
7. Exercises in Spoken English. Parts. I-III, Hyderabad. Oxford University Press.
8. Syamala, V. Speak English in Four Easy Steps, Improve English Foundation Trivandrum: 2006

**Course Outcomes (CO):**

**At the end of this course students will have:**

CO1: Ability to design a language component or process to meet desired need within realistic, Constraints such as economic, environmental, social, political, ethical, scenario

CO2: Ability to analyze the usage of English words in different contexts.

CO3: An understanding of technical and academic articles’ comprehension.

CO4: The ability to present oneself at multinational levels knowing the type of different standards of English

**Cultural Education I**

**Semester-I**

**BCA I Year**

**BCA. (common to all disciplines)-I Semester**

**Contact Hours (L-T-P): 2-0-0**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 2 | 0 | 0 | 2 |

**Course Objectives**

1. To make the students feel gratitude towards the rich religious and cultural heritage of India.
2. To understand the role of great personalities and movements in the progress of India.

**UNIT-I Holy Scriptures-A**

1. Introduction to Vedanta and Bhagavad Gita, Goals of Life – Purusharthas, Introduction to different Dhram Granthas (Various religious scriptures from Hindu, Muslim, Christian, Bodh, Jain religions)
2. Introduction to Yoga, Overview of Patanjali’s Yoga Sutras

**UNIT-II Society and Culture-I**

1. Introduction to Indian Culture and Major Symbols of Indian Culture
2. Major Indian Cultural and Ethical Values- Respect, Compassion, Kindness, Forgiveness, Introspection, Honesty, Justice, Loyalty, Devotion, Self Sacrifice, Hospitality, Vasudhev Kutumbkum

**UNIT-III India in Progress-I**

1. Education , Science and Technology in Ancient India
2. Values from Indian History- War of Mahabharta, War of Kalinga, Freedom Struggle of India, Major Farmer Movements, Major Religious and Social Upliftment Movements

**UNIT-IV Great Indian Personalities-I**

1. Life and works of the Great People of Ancient India- Sushruta, Dadhichi, Ashtvakra, Anusuya, Panini, Charaka, Kalidas, Aryabhatta, Samudragupta, Ashoka, Chandragupt Mourya, Porus, Satyabhama, Dhruv, Prahlad,Chankya,Varahmihira, Bhism, Karan, Dronacharya, MeeraBai, Surdas, Dadudayal, Kabir, Mahatma Budhha, Mahavir,Guru Nanak Dev, Guru Gobind Singh, Mohammad Saheb, Jesus Christ, Veer Shivaji, MaharanaPratap, Maharani LaxmiBai, MaharaniPadmini, Hadi Rani ShalKanwar, PannaDhai

**\***Each student shall write a detailed Report/ Critique on one topic from section -A to C and one Great Personality from Section- D leading to publication of Newspaper/ Magazine article or a review paper in a Research Journal. In addition to s/he will be required to make a Power Point Presentation on the learning and face Viva-voce by committee of teachers.

**Course Outcomes (CO):**

**At the end of this course students will have:**

CO1: Ability to acknowledge and appreciate the richness of Indian Culture

CO2: Ability to represent the culture ethics in real life

**Suggested Reading:**

1. Glory of Indian Culture (English) Paper back by [Giriraj Shah](https://www.amazon.in/-/hi/s/ref=dp_byline_sr_book_1?ie=UTF8&field-author=Giriraj+Shah&search-alias=stripbooks) 2. Historicity of Vedic and Ramayana Eras: Scientific Evidences from the Depths of Oceans to the Heights of Skies by Saroj Bala , Kulbhushan Mishra

References https://knowindia.gov.in/culture-and-heritage/lifestyle-values-and-beliefs.php

**Semester – II**

**Course Name: Database Management System**

**Course Code: BCA209**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 1 | 0 | 4 |

**Course Objectives:**

1. To enhance the fundamentals knowledge of data models and to conceptualize and depict a database system using ER diagram.
2. To develop programs and queries based on SQL and relational database, and do fundamental Operations of Relational Algebra & their Implementation.
3. To be able to know the fundamental concepts of transaction processing management.
4. To justify the concept of concurrency control techniques and recovery procedure.
5. To have an introductory knowledge about the Storage and Query processing Techniques.

**Syllabus**

**Unit I**

Basic Concepts: Purpose of database systems-Components of DBMS – DBMS Architecture and Data Independence- Data modeling, Entity Relationship Model, Relational, Network, Hierarchical and object oriented models, Data Modeling using the Entity Relationship Model.

**Unit II**

Structure of relational databases:Relational Constraints, Domain Constraints, Key Constraints Referential Integrity Constraints, Relational Algebra, Fundamental Operations of Relational Algebra & their Implementation, Data definition with SQL, insert, delete and update statements in SQL – views – data manipulation with SQL.

**Unit III**

Query Processing: Methods for Joining Tables –Nested Loop Join Merge Join, Hybrid Join, Multiple table Join, Transforming Nested Queries to Joins, Object Relational SQL, Procedural SQL, Introduction to Embedded SQL

**Unit IV**

Database Design: Integrity Constraints – Domain Constraints- Referential integrity – Functional Dependency- Normalization using Functional Dependencies, Normal forms based on primary keys- general definitions of Second and Third Normal Forms. Boyce Codd Normal Form– Multivalued Dependencies and Forth Normal Form – Join Dependencies and Fifth Normal Form.

Transaction Management: Transaction Concept, ACID Properties, Transaction State, Implementation of ACID properties, Schedules.Concurrency Control: Need of concurrency control, Concurrency control techniques, Lock based protocols, binary lock, share and exclusive lock, two phase locking protocol. Introduction to recovery.

**Unit V**

AWS DBMS: Introduction about AWS DBMS, Uses and applications of AWS DBMS, Comparison of AWS DBMS with traditional DBMS system, Introduction of AWS database services : Amazon Aurora, Amazon RDS, Amazon DynamoDB, Amazon Redshift , Amazon Neptune, Amazon DoucumentDB ( With MongoDB compatibility), Design, provision and configure an appropriate database for a given scenario.

**Course Outcomes:**

**Upon successful completion of this subject students should be able to:**

CO1:Recall the basic concept of DBMS and models of DBMS.

CO2: To know the fundamental concepts of transaction processing management.

CO3: To develop programs and queries based on SQL and relational database, and do fundamental Operations of Relational Algebra & their Implementation.

CO4: Contrast the concept of functional dependency, Normal forms, constraints and integrity

CO5: Able to understand the concept of concurrency control techniques and recovery

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | H |  | M |  |  |  |  |  | M |  |  | M |
| CO2 |  | H |  |  | H |  |  |  |  |  |  |  |
| CO3 |  |  | M | H |  |  |  | M |  |  |  |  |
| CO4 |  |  |  |  |  | M |  |  |  |  | M | H |
| CO5 | M |  |  |  |  |  | H |  |  |  |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Fundamentals of Database System Elmasri and Navathe (4rd Edition), Pearson Education Asia (2008)
2. An Introduction to Database Systems - C.J.Date (7th Edition) Pearson Education Asia (2006)

**Reference Books**

1. A.Silberschatz, H. Korth and S. Sudarshan, *Database System Concepts*, 5th Edition, McGraw Hill, 2010.
2. R. Ramakrishnan, J. Gehrke, *Database Management Systems*, 3rd edition, McGraw Hill International Edition, 2007.
3. Database System Concepts - Henry F Korth, Abraham Silbershatz, McGraw Hill 2nd edition. (2005).

**Semester – II**

**Course Name: Object Oriented Programming Using C++**

**Course Code: BCA210**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 1 | 0 | 4 |

**Course Objectives:**

1. To explain the difference between object oriented programming and procedural programming and features of object oriented programming.
2. To be able to create programs using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.
3. To be able to build C++ classes using appropriate encapsulation, objects and functions.
4. To be able to apply operator overloading in different form.
5. To be able to understand the working of files.

UNIT 1

Introduction to C++, Object Oriented Concepts, Basics of C++ environment, Classes & Object, Data members, Access specifiers, Defining member functions, inline member functions, nesting of member functions, Array within a class, Static data members, Constant members , Arrays of objects, Objects as arguments, Returning objects, Constructors, Default Constructors, Parameterized constructors, Copy constructors, Destructors, friend functions, friend classes.

UNIT 2

Compile time polymorphism, function overloading, Overloading operators, Overloading unary, Overloading binary, Overloading using friends, Overloading constructor Manipulation of strings using operators, overloading constructors, Inheritance, Base classes and derived classes, Protected members, Types, constructors in base derived classes,

UNIT 3

Run time Polymorphism, function overriding, virtual base class, Virtual functions, pure virtual function, Abstract classes, class containership. Exception handling- basics of exception handling, exception handling mechanism, throw , catch, rethrow exceptions.

UNIT 4

Fundamentals of pointers, New, Delete operators, pointer declarations, operations on pointers, passing pointers to function, passing an entire array to a function, pointers and two-dimensional arrays, array of pointers, passing functions to other functions, pointers to structures, this pointer.

UNIT 5

class templates, class templates with multiple parameters, function templates, function templates with multiple parameters, Data files -C++ stream classes, unformatted and formatted I/O operations, Opening and closing of files, File modes, File pointers and manipulation, Sequential input and output operations , Updating a file, Error handling during file operations.

**Course Outcomes:**

**Upon successful completion of this subject students should be able to:**

CO1: Understand the features of C++ supporting object oriented programming

CO2: Be able to program using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.

CO3: Be able to apply operator overloading in different form.

CO4: Understand how to apply the major object-oriented concepts to implement object oriented programs in C++, encapsulation, inheritance and polymorphism

CO5: Able to understand the working with files like opening and closing of files.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  | L | H |  |  |  | M |  | M |  |  |  |
| CO2 |  | H |  |  | L |  |  | M |  |  |  | L |
| CO3 | H |  |  |  | H | M |  |  | L |  |  |  |
| CO4 |  |  | H | H |  |  | L | M |  |  |  |  |
| CO5 |  | L |  |  |  |  |  |  | H |  | M |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Reema Thereja, ''Object Oriented Programming with C++", First Edition, Oxford University Press ,2015
2. Let Us C: Bala Guruswamy, TATA McGraw Hill.

**Reference Books**

1. Object Oriented Programming with C++, Sourav sahay, Oxford University, Sept 2012.
2. Richard F. Gilberg, Behrouz A. Forouzan, “Data structures, A Pseudocode Approach with C”, (2e), Cengage Learning India Pvt.Ltd, India, 2009.
3. Robert Kruc & Bruce Lening, “Data structures & Program Design in C”, (2e), Pearson, 2007.

**Semester – II**

**Course Name: Introduction to Cloud Architecture**

**Course Code: AWS 002A**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 2 |

**Course Objectives:**

**CO 1**. Make architectural decisions based on AWS architectural principles and best practices

**CO 2**. Use AWS services to make infrastructure scalable, reliable, and highly available

**CO 3**. Use AWS managed services to enable greater flexibility and resiliency in an infrastructure

**CO 4**. Increase performance and reduce cost of a cloud infrastructure built on AWS

**CO 5.** Use the AWS Well-Architected Framework to improve architectures that use AWS solutions

**Module 1 – Welcome to AWS Academy Cloud Architecting:** Course objectives and overview, Café business case introduction, roles in cloud computing.

**Module 2 – Introducing Cloud Architecting:** What is cloud architecting, The AWS Well-Architected Framework, Best practices for building solutions on AWS, AWS global infrastructure.

**Module 3 – Adding a Storage Layer-** The simplest architecture, Using Amazon S3, Storing data in Amazon S3, Moving data to and from Amazon S3, Choosing Regions for your architecture

**Module 4 – Adding a Compute Layer:-** Architectural need, Adding compute with Amazon EC2, Choosing an Amazon Machine Image (AMI) to launch an Amazon EC2 instance, Selecting an Amazon EC2 instance type, Using user data to configure an EC2 instance, Adding storage to an Amazon EC2 instance, Amazon EC2 pricing options, Amazon EC2 considerations.

**Module 5 – Adding a Database Layer:-** Architectural need, Database layer considerations, Amazon Relational Database Service (Amazon RDS), Amazon DynamoDB, Database security controls, Migrating data into AWS databases

**Module 6 – Creating a Networking Environment:-** Architectural need, Creating an AWS networking environment, Connecting your AWS networking environment to the internet, Securing your AWS networking environment

**Module 7 – Connecting Networks:-** Architectural need, Connecting your remote network with AWS Site-to-Site VPN, Connecting your remote network with AWS Direct Connect, Connecting virtual private clouds (VPCs) in AWS with VPC peering, Scaling your VPC network with AWS Transit Gateway, Connecting your VPC to supported AWS services.

**Module 8 – Securing User and Application Access:-** Architectural need, Account users and AWS Identity and Access Management (IAM), Organizing users, Federating users, Multiple accounts.

**Module 9 - Implementing Elasticity, High Availability, and Monitoring:-** Architectural need, Scaling your compute resources, Scaling your databases, Designing an environment that’s highly available, Monitoring.

**Module 10 – Automating Your Architecture:-** Architectural need, Reasons to automate, Automating your infrastructure, Automating deployments, AWS Elastic Beanstalk

**Module 11 – Caching Content:-** Architectural need, Overview of caching, Edge caching, Caching web sessions, Caching databases

**Module 12 – Building Decoupled Architectures:-** Architectural need, Decoupling your architecture, Decoupling with Amazon Simple Queue Service (Amazon SQS), Decoupling with Amazon Simple Notification Service (Amazon SNS), Sending messages between cloud applications and on-premises with Amazon MQ,

**Module 13 – Building Microservices and Serverless Architectures:-** Architectural need, Introducing microservices, Building microservice applications with AWS container services, Introducing serverless architectures, Building serverless architectures with AWS Lambda, Extending serverless architectures with Amazon API Gateway, Orchestrating microservices with AWS Step Functions.

**Module 14 – Planning for Disaster:-** Architectural need, Disaster planning strategies, Disaster recover patterns.

**Module 15 – Bridging to Certification:-** Certification exam resources, Additional resources.

**Semester – II**

**Course Name: Database Management System Lab**

**Course Code: BCA212**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1) Create ER diagrams for different DBMS applications

2) Implement Data Definition Language (DDL) commands in DBMS

3) Implement Data Manipulation Language (DML) and Data

4) Implement Control Language (DCL)

5) Join – Inner Join and Outer Join

6) High level language extensions with cursors

7) High level language extension with Triggers

8) Procedures and Functions

9) Implement Database design using E-R model and Normalization

10 ) Design the database model for implementing ACID properties of transaction.

11) Create AWS RDS Database Instance

12) Connecting to RDS Database on a DB Instance using the MySQL Workbench

13) Test Connection

**Semester – II**

**Course Name: Project Lab using C++**

**Course Code: BCA213**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

Class and objects- creating class, objects, private, public data members, member functions, object as array, arguments, returning objects Constructors & destructors- Default Constructors, Parameterized constructors, Copy constructors, friend functions, friend classes. Polymorphism- function overloading, operator overloading, overloading unary and binary

Pointers -operations on pointers, passing pointers to function, passing an entire array to a function, pointers and two-dimensional arrays, array of pointers, pointers to objects.

Inheritance, Single, multiple, hierarchical, multi-level, hybrid

Function overriding, virtual base class, Creation of pure virtual function

Using new and delete operator, pointer arithmetic

Data files - unformatted and formatted I/O operations, Opening and closing of files, File modes, File pointers and manipulation, Sequential input and output operations, updating a file, Error handling

Templates - class templates, class templates with multiple parameters, function templates, function templates with multiple parameters

Exception handling- basics of exception handling throw, catch, re throw exceptions.

**Semester – II**

**Course Name: Environmental Studies**

**Course Code:**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 2 | 0 | 0 | 2 |

**Course Objectives**

1. To define and use correctly the common terms of environmental science.
2. To explain what makes up the environment, how it functions, and how humans are part of it.
3. To apply the concepts and principles of environmental science to propose solutions to specific environmental problems.
4. To analysis environmental writings and predictions and their impact on subsequent developments in human relationship with the environment.
5. To evaluate the adequacy of conclusions about environmental phenomena.

**Syllabus**

**Unit I**

**Introduction and Natural Resources:** Multidisciplinary nature and public awareness, Renewable and nonrenewal resources and associated problems, Forest resources, Water resources, Mineral resources, Food resources, Energy resources, Land resources, Conservation of natural resources and human role.

**Unit II**

**Ecosystems:** Concept, Structure and function, Producers composers and decomposers, Energy flow, Ecological succession, Food chains webs and ecological pyramids, Characteristics structures and functions of ecosystems such as Forest, Grassland, Desert, Aquatic ecosystems.

**Unit III**

**Biodiversity and Conservation:** Definition, Genetic, Species, and Ecosystem diversity, Bio-geographical classification of India, Value of biodiversity at global, national, local levels, India as a mega diversity nation, Hot sports of biodiversity, Threats to biodiversity, Endangered and endemic species of India, In-situ and ex-situ conservation of biodiversity.

**Unit IV**

**Environmental Pollution:** Definition, Causes, effects and control of air pollution, water pollution, soil pollution, marine pollution, noise pollution, thermal pollution, nuclear hazards, human role in prevention of pollution, Solid waste management, Disaster management, floods, earthquake, cyclone and landslides.

**Unit V**

**Social issues and Environment:** Unsustainable to sustainable development, Urban problems related to energy, Water conservation and watershed management, Resettlement and re-habitation, Ethics, Climate change, Global warming, Acid rain, Ozone layer depletion, Nuclear accidents, holocaust, Waste land reclamation, Consumerism and waste products, Environment protection act, Wildlife protection act, Forest conservation act, Environmental issues in legislation, population explosion and family welfare program, Environment and human health, HIV, Women and child welfare, Role of information technology in environment and human health.

**Course Outcomes:**

**Upon successful completion of this subject students should be able to:**

CO1: Understand fundamental physical and biological principles that govern natural processes.

CO2: Understand fundamental concepts from the social sciences and the humanities underlying environmental thought and governance.

CO3: Integrate and apply perspectives from across the natural sciences, social sciences, and the humanities in the context of complex environmental problems.

CO4: Apply the concepts and principles of environmental science to propose solutions to specific environmental problems.

CO5: Able to understand social and environmental issues.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  | H |  |  |  |  |  | H |  | H |
| CO2 |  |  | H |  |  | H | M |  |  |  |  |  |
| CO3 |  | H |  |  |  |  |  |  |  |  |  |  |
| CO4 |  |  |  |  |  |  | H | H |  |  |  |  |
| CO5 | H |  |  |  | M |  |  |  |  |  | L |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Rajgopalan , “Environmental Studies From Crisis to Cure” , (3e), Oxford University Press.
2. Agarwal, K.C., Environmental Biology, Nidi Publication Ltd**.**, Bikaner, 2001.

**Reference Books**

1. BharuchaErach, Biodiversity of India, Mapin Publishing Pvt. Ltd., Ahmadabad, 2002.
2. Clark, R.S., Marine Pollution, Clanderson Press, Oxford, 2002.
3. Cunningham, W.P., et al., Environmental Encyclopedia, Jaico Publishing House, Mumbai, 2003.

**Semester-II**

**Cultural Education II**

**BCA I Year**

**B.CA. (common to all disciplines)-II Semester**

**Contact Hours (L-T-P): 2-0-0**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 2 | 0 | 0 | 2 |

**Course Objectives**

CO 1.To makes the students feel gratitude towards the rich religious and cultural heritage of India.

CO 2.To understands the role of great personalities and movements in the progress of India.

**UNIT-I Holy Scriptures-II**

1. Bhagavad Gita and Life Management
2. Highlights of Indian Scriptures - Major Incidents and terms from various religious scriptures including Ramayana, Mahabharta, Guru Granth Saheb, Bible, Quran, Jain Scriptures, Bodh Scriptures
3. Historicity of Ramayana and Mahabharata

**UNIT-II Society and Culture-II**

1. Indian Society: Its Strengths and Weaknesses
2. Health and Lifestyle related issues
3. Conservation of cultural heritage

**UNIT-III India in Progress-II**

1. Role & Position of Women in Indian Society- Rituals like Sati, Dakin, Kanyavadh, Pardah, Devdasi, Child Marriage, Measures of Women Empowerment including Education, Constitutional and other Rights.
2. Indian Models of Economy, Business and Management

**UNIT-IV Great Indian Personalities-II**

1. Life and works of the Great People of Modern India- Raja Ram Mohan Roy, Swami Vivekananad, Madan Mohan Malviya, Ishwarchand VidyaSagar, JyotibaPhule, HomiBhabha, B.R. Ambedkar, Mahatma Gandhi, Chandra Shekhar Aazad, Abdul Hamid, Badshah Khan, Bhagat Singh, Ashfaqullah, Vir Sawarkar, Vir Banda Bahadur, Vir Haqiqat Rai, Subhash Chandra Bose, Mother Teresa, Jagdish Chandra Basu, JRD Tata, Ratan Tata, Dada Saheb Phalke, Major Dhayan Chand, A P J Abdul Kalaam, Kailash Satyarthi, Aruna Roy, Mahasweta Devi, Udaya Kumar, Narayan Murthy, Azim Premji

**\***Each student shall write a detailed Report/ Critique on one topic from section -A to C and one Great Personality from Section- D leading to publication of Newspaper/ Magazine article or a review paper in a Research Journal. In addition to s/he will be required to make a Power Point Presentation on the learning and face Viva-voce by a committee of teachers.

**Course Outcomes (CO):**

**At the end of this course students will have:**

CO1: Ability to acknowledge and appreciate the richness of Indian Culture

CO2: Ability to represent the culture ethics in real life

**Suggested Reading:**

1. Glory of Indian Culture (English) Paperback by [Giriraj Shah](https://www.amazon.in/-/hi/s/ref=dp_byline_sr_book_1?ie=UTF8&field-author=Giriraj+Shah&search-alias=stripbooks)

2. Historicity of Vedic and Ramayan Eras: Scientific Evidences from the Depths of Oceans to the Heights of Skies by Saroj Bala , Kulbhushan Mishra

**Semester-II**

**Professional Skills**

**BCA I Year**

**BCA. (common to all disciplines)-II Semester**

**Contact Hours (L-T-P): 2-0-2**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 2 | 0 | 0 | 2 |

**Course Objectives**

1. To enhance Professional competence in reading, writing, listening and speaking.
2. Switch the approach from providing information about the language to use the language.
3. Minimize the Grammar Translation Method of ELT while trying to replace it with Direct Method.
4. Introduce Communicative Method of ELT and focusing the teaching pedagogy on the student-centered learning rather than on the teacher-centered learning.
5. Ability to master three major forms of communications which are vital in academic and professional settings namely professional presentations, interviews and group communications respectively.
6. Providing a deep insight into the techniques for delivering effective presentations, winning job interviews, and actively participating in various forms of group communication.

**Syllabus: Theory**

|  |  |
| --- | --- |
| **UNIT 1** | **Professional Grooming and Professional Culture:**  Basics of corporate culture, Dressing sense-personal hygiene, Cultural adaptability, Body language components: undesirable and desirable body language, Team-ship, Leadership, Stress and Conflict management |
| **UNIT 2** | **Advanced Grammar:** Common errors related to prepositions, articles, models , Conditionals, Determiners etc, Punctuation, Proof-reading and Editing of Documents |
| **UNIT 3** | **Composition:**, Memo, Notice, Circular, Book Review, Research Article, Reports |
| **UNIT 4** | **Vocabulary Building:** Words often misspelt, One Word Substitution, Phrasal Verbs, Idioms |
| **UNIT 5** | **Reading Comprehension:** Reading different types of documents including Passages, Reports, Technical Essays, Speeches, Research Articles, Newspaper articles, Interviews etc-Skimming and Scanning-Inference and Deduction |

**Syllabus: Lab**

|  |  |  |
| --- | --- | --- |
| **L-T-P** | **Professional Skills Lab** | **Credits 2-0-1 3** |

|  |  |
| --- | --- |
| **UNIT 1** | **Professional Grooming and Professional Culture:**  Role plays and Activities on Dressing sense-personal hygiene, Cultural adaptability, Body language components: undesirable and desirable body language, Team-ship, Leadership, Stress and Conflict management |
| **UNIT 2** | **Advanced Grammar:** Exercise Sessions forCommon errors related to prepositions, articles, models , Conditionals, Determiners etc, Punctuation, Proof-reading and Editing of Documents |
| **UNIT 3** | **Composition:**, Memo, Notice, Circular, Book Review, Research Article, Reports – Giving Assignments based on practical applications, Practice sessions on different topics |
| **UNIT 4** | **Vocabulary Building:** Words often misspelt, One Word Substitution, Phrasal Verbs, Idioms- Activities related to the appropriate use of words |
| **UNIT 5** | **Reading Comprehension:** Practice Reading Unseen Paragraphs- Finding Suitable title, Summarizing, Analyzing, Finding new words etc |

**Course Outcomes (CO):**

**At the end of this course students will have:**

CO1: Ability to design a language component or process to meet desired need within realistic, Constraints such as economic, environmental, social, political, ethical, scenario

CO2: Ability to analyze the usage of English words in professional scenario.

CO3: An understanding of technical and academic articles’ comprehension.

CO4: The ability to present oneself at multinational levels as per the demand of the corporate culture

**Suggested Readings:**

1. Felixa Eskey. Tech Talk, University of Michigan. 2005
2. Michael Swan. Practical English Usage, Oxford University Press. 2005
3. Anderson, Paul. Technical Communication: A Reader Centered Approach, V Edition, Her court, 2003.
4. Thampi, G. Balamohan. Meeting the World: Writings on Contemporary Issues. Pearson, 2013.
5. Lynch, Tony. Study Listening. New Delhi: CUP, 2008.
6. Kenneth, Anderson, Tony Lynch, Joan Mac Lean. Study Speaking. New Delhi: CUP, 2008.
7. Marks, Jonathan. English Pronunciation in Use. New Delhi: CUP, 2007.
8. Syamala, V. Effective English Communication For You (Functional Grammar, Oral and Written Communication): Emerald, 2002.

**Semester – III**

**Course Name: Data Structures and Algorithms**

**Course Code: BCA216**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 1 | 0 | 4 |

**Course Objectives:**

1. To impart the basic concepts of data structures and algorithms.
2. To understand concepts about searching and sorting techniques.
3. To understand basic concepts about stacks, queues, lists, trees and graphs.
4. To understanding about writing algorithms and step by step approach in solving problems with the help of fundamental data structures.
5. To impart the basic concepts of algorithms implementation in optimized time .

Unit 1

**Introduction and overview**: Introduction, basic terminology, elementary data organization, data structures, data structures operations, Abstract data types(ADT), Algorithms: Complexity, Time-space tradeoff.

**Preliminaries:** Introduction, Mathematical notation and functions, Algorithmic notations, control structures, complexity of algorithms, other asymptotic notation for complexity of algorithm, Sub algorithms, variables, data types.

Unit 2

**String processing:**  Introduction, Basic terminology, storing strings, character data type, string as ADT, String operations, word/text processing, Pattern matching algorithms.

**Arrays, Records and Pointers:**  Introduction, linear arrays, arrays as ADT, representation of linear arrays in memory, traversing linear arrays, inserting and deleting, sorting , bubble sort, Searching, linear search. Binary search, multidimensional arrays, representations of polynomials using arrays, Dynamic memory management, records structure, Representation of records in memory, parallel arrays, matrices and sparse matrices.

Unit 3

Linked Lists: Introduction of linked list, representation of linked list in memory traversing a linked list, searching a linked list, memory allocation garage collection, insertion into a linked list, deletion from a linked list ,header linked list, circular linked list, two way list( doubly linked lists) Buddy systems.

Stacks, queues, recursion: Introduction, stacks, array representation of stacks, linked representation of stacks, stack as ADT, Arithmetic expression , polish notation, application of stacks, recursion, tower of Hanoi, implementation of recursive procedures by stacks, queues, linked representation of queues, queues as ADT, circular queues , Dequeues, priority queues, application of queue.

Unit 4

Trees: Introduction , Binary trees, representing binary trees in memory, traversing binary trees, Traversal algorithm using stacks, Binary search tree, searching ,inserting and deleting in binary tree, balanced binary tree, AVL search tree, B-Tree (searching , insertion, deletion), B+ Tree, Red Black tree, Heap, heapsort, path length and Huffman code.

Graphs and their applications: Introduction, graph theory, sequential representation of graphs, adjacency matrix, path matrix, warshall algorithm, shortest path, linked representation of graph, operation on graphs, traversing a graph, posets, topological sorting, spanning tree .

Unit 5

Sorting and searching: Introduction, sorting, insertion sort, selection sort, merge sort, shell sort, radix sort, searching and data modification and hashing.

**Course Outcomes (COs):**

**Upon successful completion of this subject students will able**

CO1: Ability to analyse algorithms and a algorithm correctness.

CO2: Ability to implement various techniques of link list.

CO3: Ability to describe stack, queue with linked list operation.

CO4: Ability to have knowledge of tree and graphs concepts.

CO5: Ability to summarize searching and sorting techniques

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  | H |  |  |  |  |  |  | M |  |  |  |
| CO2 |  |  |  |  |  | M |  |  |  | H |  |  |
| CO3 |  |  |  | H |  |  |  | M |  |  |  |  |
| CO4 |  |  |  |  |  | M |  |  |  |  | H |  |
| CO5 | H |  |  |  | M |  |  |  |  |  |  | L |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. R. G. Dromey, “How to Solve it by Computer”, Second Edition, Prentice-Hall of India, 2002.
2. Reema Thereja,, ''Data Structure using C" , Second Edition , Oxford University Press, 2014

**Reference Books**

1. M. A. Weiss, “Data Structures and Algorithm Analysis in C”, Second Edition, Pearson Education Asia, 2002.
2. ISRD Group, “Data Structures using C”, Fifth Edition Tata McGraw Hill, 2007
3. Richard F. Gilberg, Behrouz A. Forouzan, “Data Structures – A Pseudocode” Third Edition Prentice-Hall of India, 2004.

**Semester – III**

**Course Name: Programming in Java**

**Course Code: BCA 217**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 1 | 0 | 4 |

**Course Objectives:**

1. To be able to understanding the concept of programming paradigms, basic concept of object oriented programming and features of java.
2. To demonstrate the concept of data types, literals and basic structure of java programming.
3. To enhance the practical knowledge of decision making statements and control statements.
4. To be able to use variety technologies of Java and work with different platforms.
5. To examine the life cycle of applets and packages.

**Syllabus**

**Unit I**

**The history and evolution of java:** Java lineage, Introduction of modern programming C and C++, creation of Java, C# connection, java applets, security, portability, java’s magic byte code, servlets, java buzzwords( simple, object oriented, robust, multithreaded, architecture neutral, interpreted and high performance distributed and dynamic) Evolution of java, java SE 6.

**An overview of Java:**  object oriented programming, two paradigms, abstraction, oops principles, simple program of java( Entering the program, Compiling the program) Two control statements, The if statement, for loop, using block of code, lexical issues, java class libraries.

**Data types variables and arrays:** The primitive types, integers, floating points type, characters, Booleans, literals, variables, Type conversion and casting, automatic type promotion in expressions, Arrays and their types, introduction of strings.

**Unit II**

**Operators:** Arithmetic operator, bitwise operator, relational operators, Boolean logical operators, assignment operator, operator precedence.

**Control Statements:** Java’s selection statements, iteration statements, jump statement( using break, continue and return)

**Introducing classes:** Class fundamentals , declaring objects, assigning object reference variables, introducing methods, returning a value, constructors, parameterized constructors, this keyword, garbage collection.

**Unit III**

**Methods and Classes:** Overloading methods, overloading constructors, using object as parameter, argument passing, returning objects, recursion, introducing access control, and understanding static, final, arrays revisited, introduction of nested and inner classes, exploring the string class, use of command line arguments.

**Inheritance:** Basics of inheritance, use of super class and sub class variable, creating multilevel hierarchy, method overriding, use of abstract class, the object class.

**Unit IV**

**Packages and Interfaces:**  Introduction of packages, finding packages and class path, access protection, importing packages, interfaces and their types.

**Exception handling:** Exception handling fundamentals, Exception types, uncaught exceptions, use of try and catch, multiple catch clause, nested try statements, use of throw statement, javas built in exceptions, chained exceptions, use of exceptions.

**Multithreaded programming**: The java thread model, thread priorities, synchronization, messaging, thread class and the run able interface, main thread, creating a thread, thread priorities, synchronization methods, inter thread communication, deadlock, suspending, resuming, and stopping threads, modern way to suspend, resume and stopping the threads and use of multithread.

**Unit V**

**Enumeration, auto boxing and annotations (Metadata):** Enumeration, values () and values of () methods, type wrappers, auto boxing, annotations**.**

**I/O, Applets and other topics:** I/O basics, streams, byte streams, character streams, reading console input, writing console output, print writer class, reading and writing files, applets fundamentals, transient and volatile modifiers, naïve methods.

**Generics:**  introduction of generics, generics types, generic class parameters, bounded types, wildcard arguments, generic constructor, generic interfaces, raw types and legacy code, generic class hierarchies.

**The Java Library:**  string handling, string constructors, string length, special string operations, character extraction, string comparison, searching strings and modifying a string.

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Understanding the concept of programming paradigms, basic concept of object oriented programming and features of java.

CO2: Enhance the practical knowledge of decision making statements and control statements.

CO3: Examine the life cycle of applets and packages and use the exception handling mechanism for handle the errors.

CO4: Illustrate the concept of Inheritance, Constructor, operator overloading and graphics in java.

CO5: Evaluate the concept of array, string, function, class and object.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | H |  | M |  |  | L |  |  |  | M |  |  |
| CO2 |  | H |  |  |  |  |  |  |  |  |  | M |
| CO3 |  |  | H |  |  |  |  |  |  |  | M |  |
| CO4 |  |  |  |  | M |  |  | H |  |  |  |  |
| CO5 | H |  |  | M |  |  |  |  | M |  |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Programming in Java, Sachin Malhotra, Oxford University, November 2013
2. Java One step ahead, Seth and Juneja, Oxford University, May 2017.

**Reference Books**

1. Programming with Java A Primer, E. Balaguruswamy Tata McGraw Hill Companies
2. R. Nageswara Rao, “Core Java**:** An Integrated Approach”, First Edition, DT Editorial Services, 2016.
3. Herbert Schildt, “The Complete Reference”, Ninth Edition, McGraw Hill, 2014.

Cay S. Horstmann,” Core Java”, Ninth Edition, Prentice Hall,2012.

**Semester – III**

**Course Name: Statistical and Numeric Methods**

**Course Code: BCA218**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 3 |

**Course Objectives:**

CO1: Have a necessary basic concepts of a few statistical methods

CO2: Understand how to develop the null and alternative hypothesis.

CO3: To develop the analytical capability and to impart knowledge on statistical methods.

CO4: Describe how to design experiments, carry them out and analyze the data they yield.

CO5: To select the appropriate numerical technique to solve a given problem.

Unit 1

Statistics-Introduction -Measures of average-AM-Median-Mode, Measures of dispersion and its coefficients – Range – QD – SD-MD

Unit 2

Correlation- Karl Pearson’sand Spearman’s rank correlation, Regression- regression equations, regression coefficients

Unit 3

Permutations – combinations – Probability-addition theorem, multiplication theorem, independent events, conditional probability,baye’s theorem, Probability distribution-Binomial, Poisson, Normal.

Unit 4

Interpolation- Newton’s forward &backward method- Lagrange’s Method,Curvefitting-fitting a straight line

Unit 5

Solutions of Numerical, Algebraic and transcendental methods- bisection method, Newton Raphsonmethod,Simultaneous linear equations -Gauss elimination

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to**-

CO1: Have a clear perception of the power of statistical techniques.

CO2: Demonstrate the applications of these techniques to problems drawn from industry, management and other engineering fields.

CO3: Solve algebraic and transcendental equations

CO4: Solve system of non linear equations

CO5: Interpolate, Extrapolate the data points

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | H |  | M |  |  | H |  |  | M |  |  |  |
| CO2 |  | H |  |  |  |  |  |  |  |  |  | L |
| CO3 |  |  | H |  |  |  |  |  |  |  | M |  |
| CO4 |  |  |  |  | M |  |  | H |  |  |  |  |
| CO5 | H |  |  | H |  |  |  |  | M |  |  |  |

H = Highly Related; M = Medium; L = Low

TEXT BOOKS

* P.R.Vittal-Business Mathematics and Statistics,MarghamPublications,Chennai,

REFERENCE

* H.S.Hall and S.R.Knight: Higher Algebra –AITBS Publishers India.
* M.K.Venkataraman: Numerical methods in Science and Engineering-National Publishing Company,Chennai

**Semester – III**

**Course Name: Cloud Security**

**Course Code: AWS 003A**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 4 | 0 | 0 | 3 |

**Unit 1:**

**Fundamentals of Cloud Computing and Architectural Characteristics-** Understand what is Cloud computing , Architectural and Technological Influences of Cloud Computing •,Understand the Cloud deployment models a. Public, Private, Community and Hybrid models ,Scope of Control a. Software as a Service (SaaS) b. Platform as a Service (PaaS) c. Infrastructure as a Service (IaaS) • Cloud Computing Roles •,Risks and Security Concern.

**Unit 2:**

**Security Design and Architecture for Cloud Computing-** Guiding Security design principles for Cloud Computing ,Cloud Security ,Secure Isolation , Comprehensive data protection , End-to-end access control , Monitoring and auditing , Common attack vectors and threats.

**Unit 3:**

**Data Protection for Cloud Infrastructure and Services-** Understand the Cloud based Information Life Cycle Data protection for Confidentiality and Integrity, Encryption, Data Redaction, Tokenization, Obfuscation, PKI and Key, Management, Assuring data deletion, Data retention, deletion and archiving procedures for tenant data, Data Protection Strategies.

**Unit 4:**

**Enforcing Access Control for Cloud Infrastructure based Services**- Understand the access control requirements for Cloud infrastructure Common attack vectors and threats, Enforcing Access Control Strategies, Compute, Network and Storage, Authentication and Authorization, Roles-based Access Control, Multi-factor authentication, Host, storage and network access control options, OS Hardening and minimization, securing remote access, Verified and measured boot, Firewalls, IDS, IPS and honeypots.

**Unit 5:** Security Patterns for Cloud Computing – Network Security, Identity & Access Management & Trust Secure On-Premise Internet Access Secure External Cloud Connection Cloud Denial-of-Service Protection Cloud Traffic Hijacking Protection Automatically Defined Perimeter Cloud Authentication Gateway Federated Cloud Authentication Cloud Key Management Trust Attestation Service Collaborative Monitoring and Logging Independent Cloud Auditing.

**Semester – III**

**Course Name: Project Lab in PHP**

**Course Code: BCA 219**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objectives**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1. WAP to reverse the string without using any predefined function

2. WAP to check the season of the year according to given month using control statement (Switch statement)

a. spring month 3,4,5

b. summer month 6,7,8

c. autumn month 9,10,11

d. winter month 12,1,2

3. WAP to create an array for employee names and associate another array of employee's salary and print these values using for each loop.

4. WAP that will accept an array of integers as input and output an array where for each item in the source array, the new array will perform the following operations:-

a. for even numbers divide by 2

b. for odd number multiply by 3

5. WAP to make sure a value has been supplied for a form element. For example, you want to make sure a text box hasn’t been left blank.

6. WAP to know whether an email address a user has provided is valid.

7. WAP in php to show the values in the input fields after the user hits the submit button

8. WAP create a form with one text field and submit button to find string in lower case, uppercase, sub string, position of a string and trimmed string

9. WAP to find whether a given string is subset of another given string or not

10. WAP to find square and square root of a randomly generated number

11. WAP to merge 2 associative array sort that array in :-

a. ascending order

b. descending order

c. in key sorted form

12. WAP to generate captcha code

13. WAP to create session, provide it name and id and display their values and finally delete the session.

14. WAP to in which you want to set a cookie so that your website can recognize subsequent requests from the same web browser.

15. WAP to in which you want to read the value of a cookie that you’ve previously set.

16. WAP in which You want direct access to the body of a request, not just the parsed data that PHP puts in $\_POST for you. For example, you want to handle an XML document that’s been posted as part of a web services request.

17. WAP to shows how to use validate().

18. WAP to Call ob\_start() at the top of your page and ob\_end\_flush() at the bottom. You can then intermix commands that generate output and commands that send headers.

19. WAP to send compressed content to browsers that support automatic decompression.

20. WAP to use the same HTML page to emit a form and then process the data entered into it. In other words, you’re trying to avoid a proliferation of pages that each handle different steps in a transaction.

21. WAP in which you have data that can be easily represented as key/value pairs, want to store it safely, and have very fast lookups based on those keys.

22. WAP in which you want access to a SQL database to store or retrieve information. Without a database, dynamic websites aren’t very dynamic.

23. WAP in which you want to retrieve some data from your database.

24. WAP in which you want a concise way to execute a query and retrieve the data it returns.

25. WAP in which you want to know how many rows a SELECT query returned, or you want to know how many rows an INSERT, UPDATE, or DELETE query changed.

26. WAP in which you need to make text or binary data safe for queries

**Semester – III**

**Course Name: Data structures Lab**

**Course Code: BCA220**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Topic 1: Sorting – Searching**

* Write a program to implement Bubble Sort.
* Write a program to implement selection sort.
* Write a program to implement Quick Sort.
* Write a program to implement Insertion Sort.
* Write a program to implement Merge Sort.
* Write a program to implement Binary Search.

**Topic 2: Arrays –Stacks-Recursion**

* Write and test a function that transposes a square matrix.
* Write and test a recursive function that prints all the permutations of the first n characters of a string.
* Write and test a recursive function that returns the power xn
* Write a program to implement a stack of strings (illustrate the operations push (), pop(), size(), empty() and top()).
* Write a program to show the linked implementation of the Stack class.
* Write a program to covert infix to postfix.
* Write a program to implement Towers of Hanoi using Stack. Queues-Linked-Lists
* Write a program to implement a linear list and perform the operation such as insert(), search() and delete().
* Write a program to implement a queue by adding the functions such as (i) Determine the size (ii) input queue (iii) output a queue (iv) split a queue into two queues
* Write a program to search a circular linked list with a header node.

**Topic 3: Binary Trees - Binary Tree Traversal**

* Write a program to implement Binary Search Tree.
* Priority queue implementation.
* Write a program to create a binary tree and find the height of a binary tree.
* Write a program to perform the binary tree traversals.
* Write a program to perform a deletion from a Binary Tree (using a delete () function).

**Topic 4: Graphs**

* Matrix representation of graphs
* DFS traversal
* BFS traversal

**Semester – III**

**Course Name: Project Lab using Java**

**Course Code: BCA221**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objectives:**

CO1: To understand object oriented programming concepts, and apply them in problem solving.

CO2: To learn the basics of java Console, GUI based programming and networking programming.

**Unit 1 Java Fundamentals**

* Write a program to print the following triangle of numbers  
  1  
  1 2  
  1 2 3  
  1 2 3 4  
  1 2 3 4 5
* Write a simple java application, to print the message, “Welcome to java”
* Write a program to display the month of a year. Months of the year should be held in an array.
* Write a program to assign two integer values to X and Y. Using the ‘if’ statement the output of the program should display a message whether X is greater than Y.
* Write a program to find the area of rectangle.
* Write a program to list the factorial of the numbers 1 to 10. To calculate the factorial value, use while loop. (Hint Fact of 4 = 4\*3\*2\*1)

**Unit 2 OOPs in Java**

* Write a java program to add two integers and two float numbers. When no arguments are supplied, give a default value to calculate the sum. Use function overloading.
* Write a program to perform mathematical operations. Create a class called Add Sub with methods to add and subtract. Create another class called Mul Div that extends from Add Sub class to use the member data of the super class. Mul Div should have methods to multiply and divide A main function should access the methods and perform the mathematical operations.
* Write a program with class variable that is available for all instances of a class .Use static variable declaration. Observe the changes that occur in the object’s member variable values.
* Write a java program
  + To find the area and circumference of the circle by accepting the radius from the user.
  + To accept a number and find whether the number is Prime or not
* Write a java program to create a Student class with following attributes
* Enrollment No:, Name, Mark of sub1, Mark of sub2, mark of sub3, Total Marks. Total of the three marks must be calculated only when the student passes in all three subjects. The pass mark for each subject is 50. If a candidate fails in any one of the subjects his total mark must be declared as zero. Using this condition write a constructor for this class. Write separate functions for accepting and displaying student details. In the main method create an array of three student objects and display the details.
* In a college first year class are having the following attributes
* Name of the class (BCA, BCom, MHA), Name of the staff No of the students in the class, Array of students in the class Define a class called first year with above attributes and define a suitable constructor. Also write a method called best Student() which process a first year object and return the student with the highest total mark. In the main method define a first year object and find the best student of this class
* Write a Java program to define a class called employee with the name and date of appointment. Create ten employee objects as an array and sort them as per their date of appointment. ie, print them as per their seniority.
* Create a package ‘student.fulltime .BCA‘ in your current working directory
* Create a default class student in the above package with the following attributes: Name, age, sex.
* Have methods for storing as well as displaying

**Unit 3 Exception Handling**

* Write a program to demonstrate a division by zero exception
* Write a program to create an user defined exception say Pay Out Of Bounds.
* Write a small program to catch Negative Array Size Exception. This exception is caused when the array is initialized to negative values.
* Write a program to handle Null Pointer Exception and use the “finally” method to display a message to the user.

**Units 4 and 5 GUI Programming I and II**

* Write a program which create and displays a message on the window
* Write a program to draw several shapes in the created window
* Write a program to create an applet and draw grid lines
* Write a Java program which creates a frame with two buttons father and mother.  
  When we click the father button the name of the father, his age and designation must appear.  
  When we click mother similar details of mother also appear.
* Create a frame which displays your personal details with respect to a button click
* Create a simple applet which reveals the personal information of yours.
* Write a program to move different shapes according to the arrow key pressed
* Write a java Program to create a window when we press
  + M or m the window displays Good Morning
  + A or a the window displays Good After Noon
  + E or e the window displays Good Evening
  + N or n the window displays Good Night
* Demonstrate the various mouse handling events using suitable example.
* Write a program to create menu bar and pull down menus.
* Write a program to explain the multithreading with the use of multiplication table. Three threads must be defined. Each one must create one multiplication table.
* Write a program to illustrate thread priority.
* Create a GUI program in java with the following components.
  + A frame with flow layout.
  + Add the following components on to the frame.
    - i. Two Text Field
    - ii. A button with the label display
  + Allow the user to enter data into the text field
  + When the button is clicked paint the frame by displaying the data entered in the text field
  + Allow the user to properly close the frame

**Course Outcomes (COs): Upon successful completion of this lab students should be able to do**-

CO1: Understanding of OOP concepts and basics of Java programming (Console and GUI based).

CO2: The skills to apply OOP and Java programming in problem solving.

CO3: Should have the ability to extend his/her knowledge of Java programming further on his/her own.

**Semester-III**

**Life Skills-I**

**BCA-(common to all disciplines)**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 1 | 0 | 1 | 2 |

**Course Objectives:**

CO1.To prepare the students as per the industry demands.

CO 2.Switching to Activity and Task based Teaching modules.

CO 3.To focus on the linguistic aspects in relation to life situations.

CO 4.Facilitating the aspects of behavioral skills in language.

CO 5. Ability to master three major forms of communications which are vital in academic and professional settings namely professional presentations, interviews and group communications respectively.

**Syllabus: Theory**

|  |  |
| --- | --- |
| **UNIT 1** | * Basics of Debates / Speeches / Addressing the public / Extempore/Group Discussion * Basics of Narrating and describing things |
| **UNIT 2** | * Telephonic Etiquette: Casual and formal Telephonic Communication, Telephonic Interview * CV/Resume Drafting and HR Interview advance theory * Basics of Video Interviews and Video Profiles for Job |
| **UNIT 3** | * Types of listening, advantages and disadvantages |
| **UNIT 4** | * Basics of Group Discussion, Presenting New Idea/Concept/Proposal/ Project/ Report |
| **UNIT 5** | Types of personalities, Perspective towards things, ideas, views, codes, Life skills related to Multicultural environment and emotional intelligence like- Self-confidence, Self-esteem, Self-motivation, Decision making, Resourcefulness, Risk Taking, Conflict management, Stress management, Team Buildingetc |

**Course Outcomes (CO):**

**At the end of this course students will have:**

CO1: Ability to use appropriate language while communicating with the people ranging from personal to professional settings in order to meet the desired needs of economic, environmental, social, political, ethical fields.

CO2: Ability to learn by doing it practically in the classroom.

CO3: Ability to learn by creating an environment and adapting to the environment.

CO4: The ability to prepare the students as per the need of the Multi-cultural scenario around.

CO5: Ability to improve personality, self esteem and body language.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | H |  | L |  |  | H |  |  |  | M |  |  |
| CO2 |  | H |  |  |  |  |  |  |  |  |  |  |
| CO3 | M |  | H |  |  |  |  |  |  |  |  | L |
| CO4 |  |  |  |  | M |  |  | H |  |  |  |  |
| CO5 |  | H |  | H |  |  |  |  | M |  |  |  |

H = Highly Related; M = Medium; L = Low

TEXTBOOKS

* A Communicative Grammar of English: Geoffrey Leech and Jan Svartvik. Longman, London.
* Adair J (1986) - "Effective Team Building: How to make a winning team", London, U.K: Pan Books.
* Gulati S (2006) - "Corporate Soft Skills", New Delhi, India: Rupa& Co.
* The Hard Truth about Soft Skills, by Amazon Publication.

REFERENCES

* Quantitative Aptitude, by R S Aggarwal, S Chand Publ.
* Verbal and Non-verbal Reasoning, R S Aggarwal, S Chand Publ.
* Data Interpretation, R S Aggarwal, S Chand Publ. 4. Nova GRE, KAPAL GRE, Barrons GRE books;
* Quantitative Aptitude, The Institute of Chartered Accountants of India.
* More Games Teams Play, by Leslie Bendaly, McGraw-Hill Ryerson.

**Value Education I**

**Semester-III**

***Course Code* :**

**B.CA. (common to all disciplines)-III Semester**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 1 | 0 | 0 | 1 |

**Course Objectives**

1. To give exposure to students about richness and beauty of Indian way of life. India is a country where history, culture, art, aesthetics, cuisine and nature exhibit more diversity than nearly anywhere else in the world.

2. Making students familiar with the rich tapestry of Indian life, culture, arts, science and heritage which has historically drawn people from all over the world.

**Syllabus: Theory**

Lessons from the Ramayana

Introduction to Ramayana, the first Epic in the world – Influence of Ramayana on Indian values and culture – Storyline of Ramayana – Study of leading characters in Ramayana – Influence of Ramayana outside India – Relevance of Ramayana for modern times.

Lessons from the Mahabharata

Introduction to Mahabharata, the largest Epic in the world – Influence of Mahabharata on Indian values and culture – Storyline of Mahabharata – Study of leading characters in Mahabharata – Kurukshetra War and its significance - Relevance of Mahabharata for modern times.

Lessons from the Upanishads

Introduction to the Upanishads: Sruti versus Smrti - Overview of the four Vedas and the ten Principal Upanishads - The central problems of the Upanishads – The Upanishads and Indian Culture – Relevance of Upanishads for modern times – A few Upanishad Personalities: Nachiketas, SatyakamaJabala, Aruni, Shvetaketu.

Message of the Bhagavad Gita

Introduction to Bhagavad Gita – Brief storyline of Mahabharata - Context of Kurukshetra War – The anguish of Arjuna – Counsel by Sri. Krishna – Key teachings of the Bhagavad Gita – Karma Yoga, Jnana Yoga and Bhakti Yoga - Theory of Karma and Reincarnation – Concept of Dharma – Concept of Avatar - Relevance of Mahabharata for modern times.

Life and Message of Swami Vivekananda

Brief Sketch of Swami Vivekananda’s Life – Meeting with Guru – Disciplining of Narendra - Travel across India - Inspiring Life incidents – Address at the Parliament of Religions – Travel in United States and Europe – Return and reception India – Message from Swamiji’s life.

Life and Teachings of Spiritual Masters India

Sri Rama, Sri Krishna, Sri Buddha, AdiShankaracharya, Sri Ramakrishna Paramahamsa, Swami Vivekananda.

Insights into Indian Arts and Literature

The aim of this course is to present the rich literature and culture of Ancient India and help students appreciate their deep influence on Indian Life - Vedic culture, primary source of Indian Culture – Brief introduction and appreciation of a few of the art forms of India - Arts, Music, Dance, Theatre.

**Course Outcomes (CO):**

**At the end of this course students will have:**

CO1: Ability to acknowledge and appreciate the ethical beauty of India

CO2: Ability to incorporate the values of human lives in real life applications

**Life Skills-I (Lab)**

**Semester-III**

**BCA**

**BCA. (common to all disciplines)-III Semester**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objectives**

CO 1. To give exposure to students about the group discussions, debate etc

CO 2. Making students familiar with the Etiquette of telephonic conversation and to prepare a good CV/Resume.

CO 3.To give exposure of body language in GD etc.

CO 4. To develop insight of practicing GD and how to present new concepts and idea.

CO 5. To develop ability to play a good decision maker with self esteem and confidence.

**Syllabus: Lab**

|  |  |
| --- | --- |
| **UNIT 1** | * Debates / Speeches / Addressing the public / Extempore/Group Discussion * Describing a hypothetical situation / theme / surroundings / appearance/personality traits/company/ a professional Concept/New Idea, / traits/company/ a professional Concept/New Idea, / New Project through PPT and video aids |
| **UNIT 2** | * Telephonic Etiquette: Casual and formal Telephonic Communication, Telephonic Interview * CV/Resume Drafting and HR Interview practice sessions elaborating the points as per the CV and industry demand * Video Interviews and Video Profiles for Job-Practice session for Online Interviews |
| **UNIT 3** | Listening to variety of audio/video conversations including interviews news, reports, reports, GDs, dialogues from body language, logic, wit and vocabulary perspectives |
| **UNIT 4** | Group Discussion-Practice sessions, Presenting New Idea/Concept/Proposal/ Project/ Report |
| **UNIT 5** | Activities on how to be a strong Personality, Motivation, Case studies for Resourcefulness and out of the box thinking, Role plays and Case studies on Risk taking, Self confidence and Self-esteem, Decision Making, Emotion Management, Cultural Adaptability, Multicultural Perspective towards things, ideas, views, codes etc |

**Course Outcomes: After the completion of the course the student will be able to**

CO 1 Define and Identify different life skills required in personal and professional life

CO 2 Develop an awareness of the self and apply well-defined techniques to cope with emotions and stress.

CO 3 Explain the basic mechanics of effective communication and demonstrate these through presentations.

CO 4 Take part in group discussions

CO 5 Use inculcate the risk taking capacity and a good decision maker.

**Suggested Readings:**

1. A Communicative Grammar of English: Geoffrey Leech and Jan Svartvik. Longman, London.
2. Adair J (1986) - "Effective Team Building: How to make a winning team", London, U.K: Pan Books.
3. Gulati S (2006) - "Corporate Soft Skills", New Delhi, India: Rupa& Co.
4. The Hard Truth about Soft Skills, by Amazone Publication.
5. 101 Great Answers to the Toughest Interview Questions. Ron Fry. HighBridge Company. 1996.
6. Michael Swan. Practical English Usage, Oxford University Press

**Semester – IV**

**Course Name: Advance Java**

**Course Code: BCA222**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 3 |

**Course Objectives:**

1. To be able to exhaustive coverage of advanced topics on Java from tools to enterprise Java
2. To provide ample application-based examples, with step-by-step explanations
3. To provide thorough understanding of each topic through extensive examples along with the program codes and screenshots
4. To provide relevant software installation and configuration information wherever necessary
5. To be able to work with comprises keywords, objective-type questions (with answers) and subjective-type questions for students at the end of all the chapters

**Syllabus**

Unit 1

Networking: Classes to be covered Socket, ServerSocket, IPAddress, URL connections – Swing controls – JDBC - Writing JDBC applications using select, insert, delete, update.

Unit 2

SERVLETS: Introduction to Servlets (Life cycle of servlets, Java Servlets Development Kit, creating, Compiling and running servlet). The servlet API: javax. servlet package. Reading the servlet Parameters, Reading Initialization parameter. The javax.servlet.http.

Unit 3

JAVA SERVER PAGES: Configuring Tomcat JSP/Servlet server. Brief Introduction to J2EE Architecture. Advantage of JSP technology. JSP Architecture, JSP Access Model. JSP Syntax Basic (Directions, Declarations, Expression, Scriplets, Comments) JSP Implicit Object (Out, HttpServlet Request, Http Servlet Respose, Exception Handling, Session Management.

Unit 4

Package Handling HTTP Request and Response (GET/ POST Request), Using Cookies, Session Tracking. Exception Handling.

Unit 5

Introduction to EJB – Understanding MVC – Building Controllers, models and views – Integrating hibernate with spring.

**Course Outcomes (Cos):**

**Upon successful completion of this subject students should be able to:**

CO1: Revise object oriented features of java language and develop java applet programming using various techniques.

CO2: Handling exceptions and develop multi- threaded applications.

CO3: Develop applications using collection framework and concepts of generic programming.

CO4: Develop applications using Abstract Window Toolkit

CO5: Demonstrate the connectivity with SQL through JDBC and describe the JDBC architecture.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  | H | H |  |  |  |  |  | M |  |  |  |
| CO2 |  |  | M |  |  | L |  |  |  | M |  |  |
| CO3 |  |  |  | H |  |  |  |  |  |  |  | M |
| CO4 | L |  |  |  |  |  |  | H |  |  |  |  |
| CO5 |  | M |  |  | H |  |  |  |  | L |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Sachin Malhotra, Saurabh Chaudhary, ''Programming in Java”,Second Edition, Oxford University Press , 2014.
2. Advance java programming, Oxford, Uttam Kumar Roy, April 2015.

**Reference Books**

1. Programming with Java A Primer, E.Balaguruswamy Tata McGraw Hill Companies
2. Java Programzing Language Ken Arnold Pearson
3. The complete reference JAVA2, Herbert schildt. TMH

**Semester – IV**

**Course Name: Programming in Python**

**Course Code: BCA223**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 3 |

**Course Objectives:**

1. To lineup the basics of python programming, Features, history, data types and variables.
2. To be able to solve real-world problems through python programming.
3. To be able to understand problem solving approaches, programming languages, object oriented programming.
4. To construct of Python language such as control statements, functions, strings, files, data structures.
5. To apply the concept of Classes and objects, functions and array in python.

**Syllabus**

**Unit I**

**Basics Of Python Programming:** Features of Python ,History of Python ,The Future of Python ,Writing and Executing First Python Program, Literal Constants , Numbers , Strings, Variables and Identifiers , Data Types , Input Operation, Comments , Reserved Words , Indentation.

**Unit II**

Operators and Expressions, Expressions in Python, Operations on Strings , Concatenation, Multiplication (or String Repetition), Other Data Types, Tuples , Lists , Dictionary, Type Conversion .

**Unit III**

**Decision Control, Function and Modules**: Introduction to Decision Control Statements, Selection/Conditional Branching Statements, Basic Loop Structures/ Iterative Statements, while loop, Nested Loops, The break Statement, The continue Statement, The pass Statement, The else Statement used with Loops.

**Unit IV**

Function: Function Definition, Function Call, Variable Scope and Lifetime, return statement, Lambda Functions or Anonymous Functions , Documentation Strings, Recursive Functions.

**UNIT V**

**File Handling** File : File Path, Types of Files, Opening and Closing Files, Reading and Writing Files, File Positions Renaming and Deleting Files, Directory Methods

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Outline the basics of python programming, Features, history, data types and variables.

CO2: Understanding of Python especially Decision control, function and modules.

CO3: Understanding of the file handling and Data Structure.

CO4: Be exposed to object oriented concepts in Python programming, decision controls and function.

CO5: Understand the concept of file, opening, closing, reading and writing files.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | H |  |  |  |  |  |  |  | M |  |  |  |
| CO2 | H |  | M |  |  |  |  |  |  |  |  | M |
| CO3 |  |  |  | M |  |  |  |  |  | M |  |  |
| CO4 |  |  |  |  |  |  | H |  |  |  |  |  |
| CO5 |  | M |  |  | M |  |  | H |  |  |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Python Programming, Oxford, ReemaThareja, June 2017
2. “Programming Python” by Mark Lutz and O’Reilly Media

**Reference Books**

1. “Python Testing Cookbook” by Greg L Turnquist
2. “Head First Programming” by Paul Barry and David Griffiths
3. “Python Crash Course: A Hands-On, Project-Based Introduction to Programming” by Eric Matthes.

**Semester – IV**

**Course Name: Software Engineering**

**Course Code: BCA224**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 3 |

**Course Objectives**

1. To provide a comprehensive overview of Software Engineering
2. To understand the full software development life cycle, including a thorough coverage of methods, tools, principles, and guidelines.
3. To understand software metrics, real-time software design, quality assurance, reliability, risk management, cost and schedule estimation, sizing, planning, test and integration process, technical management and human
4. To be able to work with software metrics (attributes) to measure properties of the software product as a means to evaluate and control the development process.
5. To understand the concept of planning and managing the software development.

**Syllabus**

**Unit I**

**The software problem:** Learning objectives, Cost, schedule and quality, scale and changes.

**Software Processes:** Learning objectives, process and project, component software processes, software development process models, water fall model, prototyping, iterative development, rational unified process, time boxing model, Extreme programming and agile processes, using process models in a project, project management process.

**Unit II**

**Software requirement analysis and specification:** Learning objectives, values of good SRS, requirement processes, requirement specifications, desirable characteristics of an SRS, components of SRS, structure of a requirement document, functional specification with use cases, developing use cases, other approaches for analysis( Data flow diagram, Entity relationship diagrams) and validation.

**Unit III**

**Software Architecture:** Learning objectives, role of software architecture, architecture views, components and connector views, Architecture styles for C&C view, pipe and filters, shared data style, client server style, documenting architecture design, evaluating architecture.

**Planning software project:** Learning objectives, effort estimation, top down estimation approach, bottom up estimation approach, project schedule and staffing, quality planning, risk management planning, risk assessment, risk control, risk management planning approach, project monitoring plan, measurement, project monitoring and tracking and detailed scheduling.

**Unit IV**

**Design:** Learning objectives , design concepts, coupling, cohesion, open cloud principle, function oriented design, structure charts, structure design methodology, object oriented design, OO concepts, Unified modeling language(UML) Design methodology, verification, Metrics( Complexity metrics for function oriented design, complexity metrics for OO design.

**Coding and unit testing:** Learning objectives, Programming principles and guidelines, structured programming, information hiding, some programming practices, coding standards, incremental developing code and coding process, Test driven development, pair programming, managing evolving code, source code control and build, refactoring, unit testing, testing procedural units, unit testing of classes, code inspection, planning, self review, group review meeting, metrics, size measures, complexity metrics.

**Unit V**

**Testing :** Learning objectives, testing concepts, error, fault and failure, test case, test suite and test harness, psychology of testing, levels of testing, testing process, test plan, test case design, test case execution, Black box testing, equivalence class partitioning, boundary value analysis, pairwise testing, special cases, state-based testing, white box testing, control flow-based criteria, test case generation and tool support, metrics, coverage analysis, reliability, defect removal efficiency.

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Understanding of Basics of Software Engineering & Development.

CO2: Understanding of Different-2 models, Software Matrices and Estimation.

CO3: Understanding of Software Development across Workflows and Phases, languages & Architecture.

CO4: Understanding of different -3 testing strategies and tools.

CO5: Be exposed to World Wide Web and Enterprise Software Development.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcomes** | **Program Outcomes** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | H |  |  | M |  |  |  | M |  |  |  |  |
| CO2 |  | M |  |  |  | M |  |  |  |  | M |  |
| CO3 | H |  |  |  |  |  | M |  |  |  |  |  |
| CO4 |  |  | H |  |  |  |  |  | M |  |  | L |
| CO5 | M |  |  |  |  |  |  |  |  |  |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Software Engineering Fundamentals, Oxford, Ali Behforooz and Frederick Hudson, October 2012.
2. Pressman, R.S., ''Software engineering" A Practitioner's Approach", Third Edition

**Reference Books**

1. Jalote P., "An Integrated approach to Software Engineering", Narosa,1991.
2. Ghezzi Carlo and Others "Fundamentals of Software Engineering" PHI
3. Fairley R., "Software Engineering Concepts", Tata McGraw Hill, 1997.

**Semester – IV**

**Course Name: Web Technologies**

**Course Code: BCA225**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 3 |

**Course Objectives:**

1. To be able to develop a dynamic webpage by the use of java script and DHTML.
2. To be able to write a well formed / valid XML document.
3. To be able to connect a java program to a DBMS and perform insert, update and delete operations on DBMS table.
4. To be able to write a server side java application called Servlet to catch form data sent from client, process it and store it on database.
5. To be able to write a server side java application called JSP to catch form data sent from client and store it on database.

**Syllabus**

**Unit -1**

HTML5 and CSS3 HTML5- Basic Tags, Tables,Forms.HTML5 Tags,HTML Graphics, HTML media, HTML Graphics,HTML APIs. CSS - Background, Borders,margin, Box model. Styling text, fonts, list, links, tables. CSS overflow, float ,inline blocks, pseudo classes, pseudoelements.CSS border images, rounded corners

**Unit-2**

Java Script Client side scripting using java script, Introduction to java script, internal and external Java script files, variables, control statements, loops, Arrays , string handling , How to write functions in JavaScript, inputting and outputting from form elements to JavaScript. DOM concept, creating html elements using java script. Drawing 2D shapes, handling events. Introduction to AJAX

**Unit-3**

Building Single page applications with Angular JS Single page application – introduction , two way data binding, MVC in angular JS, controllers, getting user inputs , loops , Client side routing – accessing URL data , various ways to provide data in angular JS.

**Unit -4**

Server Side Programming Server side scripting, Difference between client side and server side scripting languages. Introduction to PHP, variables, control statements, loops, Arrays, string handling, PHP forms, Global variables in PHP, Regular expression and pattern matching, Database programming: inputting and outputting data from MySQL using PHP, insertion , deletion and updating data. State management in web applications, cookies, Application and session state.

**Unit-5**

Introduction to Xml, usage of XML, XML tags, elements and attributes, attribute type, XML validation: DTD and XSD, XML DOM Case study:-Application Development using Laravel framework

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Students are able to understand basic terms and protocols related to WWW.

CO2: Students are able to develop a dynamic webpage by the use of HTML/DHTML

CO3: Students will be able to write a well formed / valid XML document.

CO4: Students will be able to use java script for event handling etc.

CO5: Students will be able to write a server side java application called Servlet to catch form data sent from client, process it and store it on database.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  | L | H |  |  |  | M |  | M |  |  |  |
| CO2 |  | H |  |  | L |  |  | M |  |  |  | L |
| CO3 | H |  |  |  | H | M |  |  | L |  |  |  |
| CO4 |  |  | H | H |  |  | L | M |  |  |  |  |
| CO5 |  | L |  |  |  |  |  |  | H |  | M |  |

H = Highly Related; M = Medium; L = Low

**Text books and references**

The Complete Reference, HTML and CSS by Thomas A Powell latest edition

XML Bible by Horold, Ellotte Rusty

Web Reference:- W3Schools.com

**Semester – IV**

**Course Name: AWS Academy Cloud Foundation Course**

**Course Code: AWS 004A**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 3 |

**Course Objectives** : Upon completion of this course, students will be able to

CO 1 Define the AWS Cloud

CO 2 Explain the AWS pricing philosophy

CO3 Identify the global infrastructure components of AWS

CO 4 Describe the security and compliance measures of the AWS Cloud, including AWS Identity and Access Management (IAM)

CO 5Create a virtual private cloud (VPC) by using Amazon Virtual Private Cloud (Amazon VPC)

**Module 1**. **Cloud Concepts Overview:**

Introduction to cloud computing, Advantages of the cloud, Introduction to AWS, Moving to the AWS Cloud.

**Module 2**. **Cloud Economic and Billing:**

Introduction, Fundamentals of Pricing, Total cost of ownership, Delaware North Case study, AWS organization, AWS Billing and cost Management, Billing dashboard, Technical Support Model,

**Module 3**. **AWS Global Infrastructure Overview:**

Introduction, AWS Global Infrastructure, AWS Services and Service category, AWS management console click through.

**Module 4:**

Introduction, AWS shared responsibility model, AWS IAM, Securing a New AWS Account, Securing account and data, working to ensure compliance

**Module 5:**.**Networking and Content Delivery:**

Introduction, Networking Basics, Amazon VPC, VPC Networking, VPC Security, Route 53, Cloud front,

**Module 6. Compute:** Introduction, compute services overview, amazon EC2 part 1, amazon EC2 part 2, amazon EC2 part 3,amazon EC2 cost optimization, container service, Introduction to AWS Lambda, Introduction to AWS Elastic Beanstalk,

**Module 7. Storage:**

Introduction, AWS EBS( Elastic Block Store Console), AWS S3, AWS EFS, AWS S3 Glacier.

**Module 8. Databases:**

Introduction , Amazon RDS, Amazon DynamoDB, Amazon Redshift, Amazon Aurora.

**Module 9. Cloud Architecture:**

Introduction, AWS Well Architected framework design principles, operational excellence, security, reliability, performance efficiency, cost optimization, reliability and high availability, AWS trusted advisor,

**Module 10. Automatic Scaling Monitoring:**

Introduction, Elastic Load Balancing, Amazon Cloud watch, Amazon EC2 Auto Scaling,

**Semester – IV**

**Course Name: Project Lab (Web Technologies)**

**Course Code: BCA226**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 3 |

**Course Objectives:**

CO1. Able to build a static website using HTML

CO2. Able to include JavaScript for validations

CO3. Able to use XML to store and forwarding data.

CO4. Students able to implement dynamic websites using HTML

CO5. Able to develop Web applications by using JSP with Database Connectivity.

1. Create a web page with advanced layouts and positioning with CSS and HTML.
2. Design a website with different methods of embedding CSS in a web page.
3. Create a static web page which displays your personal details. (Hint: CSS3 and HTML5)
4. Create a web page through which the user can enter his / her details to become an authenticated user of that page.
5. Create a web site for a Computer Hardware shop. (Hint: CSS3 and HTML5)
6. Create a web site for Amrita School of Arts and Sciences. (Hint: CSS3 and HTML5)
7. Create a web page that shows different methods of embedding JavaScript.
8. Create a web page with rollover menus. Rollover menus should be created using JavaScript.
9. Create a simple calculator, which can perform the basic arithmetic operations.
10. Validate the registration for with the following criteria:
    * Name and Age should be Mandatory Fields.
    * Password and Re-enter Password fields should contain same value.
    * Name field should accept only character values.
11. Write a PHP program to store current date-time in a COOKIE and display the ‘Last visited on’ date-time on the web page upon reopening of the same page.
12. Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.
13. Using PHP and MySQL, develop a program to accept book information viz. Accession number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings.
14. Create a registration form using Angular JS.
15. Create a simple AngularJS calculator application using Angular Services.
16. Create an application Searching for a character and displaying its position using AngularJS.
17. Create an application using angular JS filters.
18. Create single page web applications using the MVC pattern of AngularJS.
19. Design an XML document to store information about a student in an engineering college affiliated to Amrita. The information must include USN, Name, Name of the College, Branch, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.
20. Create an XML document with the following sample real estate data
    * Root element real-estate will contain a sequence of sub-elements agencies, owners, properties and flats, all with an empty content
    * Ensure well-formedness
21. Create an internal DTD for the previous XML document
    * Ensure its validity
    * Then try to break it
22. Move the previous DTD to an external file and validate the XML document again
23. Create an application that loads a text string into an XML DOM object, and extracts the info from it with JavaScript.
24. Create an application which reads data from an XML file into XMLDOM object and retrieves the text value of the first element in the xml file.

***Course Outcome (CO):***

At the end of this course students will have:

CO1: Able to understand the basics of computer network, various protocols

CO2: Ability to understand WWW and HTML language

CO3: Ability to develop projects by formatting HTML documents &managing images in HTML

CO4: Able to understand Hypertext and Link in HTML

CO5: Ability to understand CSS programming language

TEXT BOOKS

1. Deitel H.M. and Deitel P.J., “Internet and World Wide Web How to program”, Pearson International, 2012, 4th Edition.

2. Gopalan N.P. and Akilandeswari J., “Web Technology”, Prentice Hall of India, 2011Paul Dietel and Harvey Deitel,”Java How to Program”, Prentice Hall of India, 8th Edition C

**Semester – IV**

**Course Name: Project Lab in Advance java**

**Course Code: BCA227**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objectives:**

CO1. Able to build a static website using Swing components

CO2. Able to include JavaScript for validations

CO3. Able to use servlets and JSP.

CO4. Students able to implement dynamic websites using HTML

CO5. Able to develop Web applications by using JSP with Database Connectivity.

* Program to demonstrate Swing components.
* Program to implement Address Book using Swing components.
* Program to demonstrate loading of file in an Swing Component.
* Multithreading program, one of the threads print a….z and other thread print 1…26.
* Example: 1a2b3c…. 26z.
* Multithreading program to schedule two jobs.
* Client Server Socket Programming.
* Server Socket which receives data from a java client program using JSON
* Program to fetch a particular Website tags when an URL is specified.
* Implement stack, queue, hashmap, hashtable, enumeration, ArrayList.
* Create a table from a java program.
* Update a table from a java program.
* Load a table data in Swing components.
* Delete a record from a table, drop table from a java file.
* Program which shows use of Statement, Prepared Statement and Callable Statement.
* Configure Apache Tomcat and write a hello world jsp page.
* Configure Apache Tomcat server to deploy Servlets.
* Exceptional handling in a JSP page.
* Create a login page and authenticate a user in a JSP page using database.
* Write a program to implement a simple servlet which writes a Welcome HTML page in the web browser.
* A servlet should receive a parameter from JSP page and process it.
* Servlet program to implement parameter handling.
* Servlet program to handle GET and POST request.
* A website hit counter data which has to be saved in a cookie.
* Implement a Java Beans to set and get values.
* Program to illustrate the procedure of handling session and print a Hello world using Java Bean.
* Enterprise Session Beans, deploy, and run a simple Java EE application which does add, subtract, multiply and division using stateless session bean.
* An application named account using stateful session bean. The purpose of account is to perform transaction operations (deposit and withdraw) for the customer.
* The account application consists of an enterprise bean, which performs the transactions, and two types of clients: an application client and a web client low the user to properly close the frame

**Course Outcome:**

At the end of this course student will:

CO1 Interpret the need for advanced Java concepts like enumerations and collections in developing modular and efficient programs

CO2 Build client-server applications and TCP/IP socket programs

CO3 Describe the working of string methods

CO4 Illustrate database access and details for managing information using the JDBC API

CO5 Describe how servlets fit into Java-based web application architecture.

**Semester – IV**

**Course Name: Project Lab in Python**

**Course Code: BCA228**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objectives:**

CO1. Understand the basic building blocks in python programming language to construct different applications

CO2. Apply the necessary data structures to solve a given problem.

CO3. Extract and import packages for developing different solutions for real time problems.

CO4. Implement the problems in terms of real -world objects using concept of OOPS.

CO5.Apply basic problem solving techniques using python.

**Experiment 1.** [Arithmetic Operations](https://python-iitk.vlabs.ac.in/exp/arithmetic-operations/)

**Experiment 2.** [Built-in Functions](https://python-iitk.vlabs.ac.in/exp/built-in-functions/)

**Experiment 3.** [Loops](https://python-iitk.vlabs.ac.in/exp/loops/)

**Experiment 4.** [Data Types](https://python-iitk.vlabs.ac.in/exp/data-types/)

**Experiment 5.**[Strings](https://python-iitk.vlabs.ac.in/exp/strings/)

**Experiment 6** [Classes and Objects](https://python-iitk.vlabs.ac.in/exp/classes-and-objects/)

**Experiment 7.** [Built-in Modules](https://python-iitk.vlabs.ac.in/exp/built-in-modules/)

**Experiment 8 .** [Constructors and Inheritance](https://python-iitk.vlabs.ac.in/exp/constructors-and-inheritance/)

**Experiment 9.** [File Operators](https://python-iitk.vlabs.ac.in/exp/file-operators/)

**Course Outcome:**

At the end of this course student will:

CO1 Explain the syntax for python programming constructs.

CO2 Compute the compound data using Python lists, tuples, and dictionaries

CO3 Examine the programs with sub problems using ‘Python’ language.

CO4 Apply python programs to read and write data from/to files .

CO5 Compute the flow of the program to obtain the programmatic solution.

**Semester – IV**

**Course Name: Life skill –II (Aptitude)**

**Course Code:**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 1 | 0 | 0 | 1 |

**Course Objectives:**

CO1. To acquire the basic professional skills used in business and to learn etiquettes and socially accepted behavior.

CO2. Apply the concept of GD with a clear and open discussion on any relevant topics

CO3. To learn the basic concepts of reading comprehension and problem solving techniques.

CO4. To acquire the Data Sufficiency: Concepts and Problem Solving.

CO5. Learn the concepts of reasoning with special focus on Non-verbal techniques.

Professional Grooming and Practices: Basics of Corporate culture, Key pillars of Business Etiquette. Basics of Etiquette: Etiquette – Socially acceptable ways of behavior, Personal hygiene, Professional attire, Cultural Adaptability. Introductions and Greetings: Rules of the handshake, Earning respect, Business manners. Telephone Etiquette: activities during the conversation, Conclude the call, To take a message. Body Language: Components, Undesirable body language, Desirable body language. Adapting to Corporate life: Dealing with people.

Group Discussions: Advantages of Group Discussions, Structured GD – Roles, Negative roles to be avoided, Personality traits to do well in a GD, Initiation techniques, How to perform in a group discussion, Summarization techniques. Listening Comprehension advanced: Exercise on improving listening skills, Grammar basics: Topics like clauses, punctuation, capitalization, number agreement, pronouns, tenses etc.

Reading Comprehension advanced: A course on how to approach middle level reading comprehension passages.  
Problem solving – Money Related problems; Mixtures; Symbol Based problems; Clocks and Calendars; Simple, Linear, Quadratic and Polynomial Equations; Special Equations; Inequalities; Functions and Graphs; Sequence and Series; Set Theory; Permutations and Combinations; Probability; Statistics.  
Data Sufficiency: Concepts and Problem Solving.  
Non-Verbal Reasoning and Simple Engineering Aptitude: Mirror Image; Water Image; Paper Folding; Paper Cutting; Grouping Of Figures; Figure Formation and Analysis; Completion of Incomplete Pattern; Figure Matrix; Miscellaneous.  
Special Aptitude: Cloth, Leather, 2D and 3D Objects, Coin, Match Sticks, Stubs, Chalk, Chess Board, Land and geodesic problems etc., Related Problems

Course Outcomes: After the completion of the course the student will be able to

CO 1 Define and Identify different life skills required in personal and professional life

CO 2 Develop an awareness of the self and apply well-defined techniques to cope with emotions and stress.

CO 3 Explain the basic mechanics of effective communication and demonstrate these through presentations.

CO 4 Take part in group discussions

CO 5 Use appropriate thinking and problem solving techniques to solve new problems.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  | L | H |  |  |  | M |  | M |  |  |  |
| CO2 |  | H |  |  | H |  |  | M |  |  |  | L |
| CO3 | H |  |  |  | H | M |  |  | L |  |  |  |
| CO4 | H |  | M | H |  |  | L |  | H |  |  |  |
| CO5 |  | L |  |  | M |  |  |  | H |  | M |  |

H = Highly Related; M = Medium; L = Low

TEXTBOOKS

* A Communicative Grammar of English: Geoffrey Leech and Jan Svartvik. Longman, London.
* Adair J (1986) - "Effective Team Building: How to make a winning team", London, U.K: Pan Books.
* Gulati S (2006) - "Corporate Soft Skills", New Delhi, India: Rupa& Co.
* The Hard Truth about Soft Skills, by Amazone Publication.

REFERENCES

* Quantitative Aptitude, by R S Aggarwal, S Chand Publ.
* Verbal and Non-verbal Reasoning, R S Aggarwal, S Chand Publ.
* Quantitative Aptitude by AbjithGuha, Tata McGraw hill Publ.
* More Games Teams Play, by Leslie Bendaly, McGraw-Hill Ryerson.
* The BBC and British Council online resources
* Owl Purdue University online teaching resources

**Semester-IV**

**Value Education and Ethics II**

**BCA II Year**

**B.CA. (common to all disciplines)-IV Semester**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 1 | 0 | 0 | 1 |

**Course Objectives**

1. To give exposure to students about richness and beauty of Indian way of life. India is a country where history, culture, art, aesthetics, cuisine and nature exhibit more diversity than nearly anywhere else in the world.

2. Making students familiar with the rich tapestry of Indian life, culture, arts, science and heritage which has historically drawn people from all over the world.

**Yoga and Meditation**

The objective of the course is to provide practical training in YOGA ASANAS with a sound theoretical base and theory classes on selected verses of Patanjali’s Yoga Sutra and Ashtanga Yoga. The coverage also includes the effect of yoga on integrated personality development. **Rajasthan Mural Art and Painting**

Mural painting is an offshoot of the devotional tradition in Rajasthan. A mural is any piece of artwork painted or applied directly on a wall, ceiling or other large permanent surface. In the contemporary scenario Mural painting is not restricted to the permanent structures and are being done even on canvas. Rajasthani mural paintings are the frescos depicting mythology and legends, which are drawn on the walls of temples, principally in Rajasthan. Ancient temples and tourists places in different States of Rajasthan, display an abounding tradition of mural paintings mostly dating back between the 9th to 12th centuries when this form of art enjoyed Royal patronage. Learning Mural painting through the theory and practice workshop is the objective of this course.

**Course on Organic Farming and Sustainability**

Organic farming is emerging as an important segment of human sustainability and healthy life. Haritamritam’ is an attempt to empower the youth with basic skills in tradition of organic farming and to revive the culture of growing vegetables that one consumes, without using chemicals and pesticides. Growth of Agriculture through such positive initiatives will go a long way in nation development. It is a big step in restoring the lost harmony of nature.

**Benefits of Indian Medicinal Systems**

Indian medicinal systems are one of the most ancient in the world. Even today society continues to derive enormous benefits from the wealth of knowledge in Ayurveda of which is recognised as a viable and sustainable medicinal tradition. This course will expose students to the fundamental principles and philosophy of Ayurveda and other Indian medicinal traditions.

**Traditional Fine Arts of India**

India is home to one of the most diverse Art forms world over. The underlying philosophy of Indian life is ‘Únity in Diversity” and it has led to the most diverse expressions of culture in India. Most art forms of India are an expression of devotion by the devotee towards the Lord and its influence in Indian life is very pervasive. This course will introduce students to the deeper philosophical basis of Indian Art forms and attempt to provide a practical demonstration of the continuing relevance of the Art.

**Science of Worship in India**

Indian mode of worship is unique among the world civilisations. Nowhere in the world has the philosophical idea of reverence and worshipfulness for everything in this universe found universal acceptance as it in India. Indian religious life even today is a practical demonstration of the potential for realisation of this profound truth. To see the all-pervading consciousness in everything, including animate and inanimate, and constituting society to realise this truth can be seen as the epitome of civilizational excellence. This course will discuss the principles and rationale behind different modes of worship prevalent in India

**Insights into Indian Classical Music**

The course introduces the students into the various terminologies used in Indian musicology and their explanations, like Nadam, Sruti, Svaram – svara nomenclature, Stayi, Graha, Nyasa, Amsa, Thala,- Saptatalas and their angas, Shadangas, Vadi, Samavadi, Anuvadi. The course takes the students through Carnatic as well as Hindustani classical styles.

**Insights into Traditional Indian Painting**

The course introduces traditional Indian paintings in the light of ancient Indian wisdom in the fields of aesthetics, the Shadanga (Six limbs of Indian paintings) and the contextual stories from ancient texts from where the paintings originated. The course introduces the painting styles such as Madhubani, Kerala Mural, Pahari, Cheriyal, Rajput, Tanjore etc.

**Insights into Indian Classical Dance**

The course takes the students through the ancient Indian text on aesthetics the Natyasastra and its commentary the AbhinavaBharati. The course introduces various styles of Indian classical dance such as Bharatanatyan, Mohiniyatton, Kuchipudi, Odissy, Katak etc. The course takes the students through both contextual theory as well as practice time.

**Indian Martial Arts and Self Defense**

The course introduces the students to the ancient Indian system of self-defense and the combat through various martial art forms and focuses more on traditional Kerala’s traditional KalariPayattu. The course introduces the various exercise technique to make the body supple and flexible before going into the steps and techniques of the martial art. The advanced level of this course introduces the technique of weaponry.

**Social Awareness Campaign**

The course introduces the students into the concept of public social awareness and how to transmit the messages of social awareness through various media, both traditional and modern. The course goes through the theoretical aspects of campaign planning and execution.

**Organic Farming in Practice**

Organic agriculture is the application of a set of cultural, biological, and mechanical practices that support the cycling of farm resources, promote ecological balance, and conserve biodiversity. These include maintaining and enhancing soil and water quality; conserving wetlands, woodlands, and wildlife; and avoiding use of synthetic fertilizers, sewage sludge, irradiation, and genetic engineering. This factsheet provides an overview of some common farming practices that ensure organic integrity and operation sustainability.

**Ayurveda for Lifestyle Modification**

Ayurveda aims to integrate and balance the body, mind, and spirit which will ultimately leads to human happiness and health. Ayurveda offers methods for finding out early stages of diseases that are still undetectable by modern medical investigation. Ayurveda understands that health is a reflection of when a person is living in harmony with nature and disease arises when a person is out of harmony with the cycles of nature. All things in the universe (both living and non-living) are joined together in Ayurveda. This leaflet endow with some practical knowledge to rediscover our pre- industrial herbal heritage.

**Life Style and Therapy using Yoga**

Yoga therapy is the adaptation of yogic principles, methods, and techniques to specific human ailments. In its ideal application, Yoga therapy is preventive in nature, as is Yoga itself, but it is also restorative in many instances, palliative in others, and curative in many others. The therapeutic effect comes to force when we practice daily and the body starts removing toxins and the rest is done by nature.

**Course Outcomes (CO):**

**At the end of this course students will have:**

CO1: Ability to acknowledge and appreciate the ethical beauty of India

CO2: Ability to incorporate the values of human lives in real life applications

**Semester – V**

**Course Name: Computer Network**

**Course Code: BCA229**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 3 |

**Course Objectives**

1. To be familiar with the terminology and concepts of Layering, Distributed Systems and Networks, Peer-to-Peer and Client-Server Networks.
2. To enhance the practical knowledge of protocols used in different layers.
3. To be familiar with physical layer based on telephone lines.
4. To be able to understand Error Detection, Error Correction, Flow Control.
5. To be able to understand the concept of Connection Oriented, Connectionless and routing algorithms.

**Syllabus**

**Unit I**

**Introduction to Computer Networks**: Definition: Network , The Need of Resources Sharing, data communications components, data representation, and data flow, network topologies, categories of networks, organizations that set standards in data communications and networking, introduction to Open Systems Interconnection (OSI) and the Internet model (TCP/IP) layers and services.

**Unit II**

**The Physical Layer & Media:** Functions of physical layer, issues related to the physical layer and the transmission medium that is controlled by the physical layer, Bandwidth Utilization: Multiplexing and Spreading, Transmission Media, Switching.

**Unit III**

**The Data Link Layer:** Error Detection and Correction, Data Link Control, Multiple Access. Wired LANs: Ethernet, Wireless LANs.

**Unit IV**

**The Network Layer:** Logical addressing: IPv4 and IPv6, Internet Protocol: IPv4, Datagram, Fragmentation, Checksum, IPv6 Advantages and Packet Format, Address mapping: ARP, RARP and DHCP, Forwarding, and Unicast and Multicast Routing.

**Unit V**

**Transport Layer and Application Layer:** Process-to-Process Delivery: UDP, TCP, and SCTP. Congestion Control. Application Layer: Domain Name System, Remote Logging, Electronic Mail, and File Transfer, WWW and HTTP.

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: To be familiar with the terminology and concepts of Layering, Distributed Systems and Networks, Peer-to-Peer and Client-Server Networks.

CO2: Describe, analyze and compare Physical Layer based on telephone lines.

CO3: Describe, analyze and compare a number of data link, network, and transport layer protocols, Error Detection, Error Correction and Flow Control.

CO4: Able to understand the concept of Connection Oriented, Connectionless and routing algorithms.

CO5: Enhance the practical knowledge of protocols used in different layers

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | H |  |  |  |  |  |  |  |  | M |  |  |
| CO2 |  | H | M |  |  |  |  |  |  |  |  | M |
| CO3 |  |  |  |  | M |  |  |  |  |  | M |  |
| CO4 |  |  |  |  |  |  | M |  |  |  |  |  |
| CO5 | M |  |  | H |  |  |  |  | M |  |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Behrouz A. Forouzan, Data Communication and Networking, Fifth Edition, Mcgraw Hill, 2017.
2. Andrew S. Tanenbaum, David J. Wetherall, Computer Networks, 5th Edition, Pearson, 2011.

**Reference Books**

# Larry L. Peterson and Bruce S. Davie, Computer Networks: A System Approach, Fifth Edition, The Morgan Kaufmann Series in Networking, 2011

1. James Kurose and Keith Ross, Computer networking: A Top Down Approach, Seventh Edition, Pearson, 2017.
2. William Stallings, Data and Computer Communications, 10th Edition, Pearson, 2014

**Semester – V**

**Course Name: Cryptography and Cyber Security**

**Course Code: BCA230**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 3 |

**Course Objectives**

CO1. To understand basics of Cryptography and Network Security.

CO2. To learn about how to maintain the Confidentiality, Integrity and Availability of a data.

CO3. To understand various protocols for network security to protect against the threats in the networks.

CO4. To know about various encryption techniques such as digital signatures use.

CO5. To study about message authentication and cyber crime security.

Unit 1

Introduction to Cyber Security - Types of Attacks, Goals for Security, Security threat and vulnerability, Cyber security models (the CIA triad, the star model). Classical encryption techniques substitution ciphers and transposition ciphers, cryptanalysis, steganography, Stream and block ciphers - Modern Block Ciphers: Block ciphers principles, Shannon’s theory of confusion and diffusion. Data encryption standard (DES), Strength of DES, Idea of differential cryptanalysis, block cipher modes of operations.

Unit 2

Principals of public key crypto systems, RSA algorithm, security of RSA. Key Management and distribution: Symmetric key distribution, Diffie-Hellman Key Exchange, Public key distribution, Introduction to SSL.

Unit 3

Message Authentication Codes: Authentication requirements, authentication functions, message authentication code, hash functions, birthday attacks, security of hash functions,

Unit 4

Digital Signatures: Digital Signatures, Elgamal Digital Signature Techniques, Digital signature standards (DSS), proof of digital signature algorithm.

Unit 5

Introduction to Cyber Crime and security: Cyber Crimes, types of Cyber Crime, hacking, attack vectors, Cross Site Scripting (XSS), XSS Consequences. Cyber Space and criminal behaviour, traditional problems associated with Cyber Crime, Introduction to Incident Response, Digital Forensics - Phishing.

**Upon successful completion of this subject students should be able to:**

CO1: Classify the symmetric encryption techniques.

CO2: Illustrate various Public key cryptographic techniques.

CO3: Summarize the intrusion detection and its solutions to overcome the attacks.

CO4: Evaluate the authentication and hash algorithms.

CO5: Basic concepts of system level security.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | M |  |  |  |  |  |  |  |  | M |  |  |
| CO2 |  | H | M |  |  |  |  |  |  |  |  | M |
| CO3 |  |  |  |  | L |  |  |  |  |  | H |  |
| CO4 |  | H |  |  |  |  | M |  |  | L |  |  |
| CO5 | H |  |  | H |  |  |  |  | M |  |  |  |

H = Highly Related; M = Medium; L = Low

**TEXTBOOK**

* William Stallings, “Cryptography and Network Security: Principals and Practice”, Pearson Education, Sixth Edition.

**REFERENCE**

* Nina God bole and Sunit Belpure, Cyber Security: Understanding Cyber crimes, Computer Foreinsics and Legal Perspectives, Willey India Pvt.Ltd.
* Dr T R Padmanabhan N Harini,”Cryptography and Security Paperback”, Wiley India

**Semester – V**

**Course Name: Mobile Application Development**

**Course Code: BCA231**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 3 |

**Course Objectives:**

1. To understand the Android application architecture, including the roles of the task stack, activities, and services.
2. To be able to build user interfaces with fragments, views, form widgets, text input, lists, tables, and more.
3. To be able to use advanced UI widgets for scrolling, tabbing, and layout control.
4. To be able to store application data on the mobile device, in internal or external storage locations.
5. To be able to support user-specific preferences using the Android Preferences API.

**Unit I**

**Overview of mobile App and mobile interface:** Mobile system, mobile interface and applications, optimizations in mobile systems, mobile embedded system, mobile cloud, big data application in mobile systems, data security and privacy protections in mobile systems, concepts of mobile apps, brief introduction of android and its framework, brief history of android, android device distribution, android sdk.

**Quick start on android:** Installing java, installing integrate development environment, installing android sdk, creating an android application, android virtual device.

**Unit II**

**Introduction of key concepts of android:** App components, activities, services, content providers, intents, App resources, App manifest( elements, attributes, declaring class names, multiple values, resource values, sting values

**Graphics and multimedia in android:** Introduction of 2-D graphics techniques, color, paint, path, canvas, draw able, button selector, Advanced UI design, overview of multimedia in android, audio implementations in android, executing video in android.

**Unit III**

**Mobile embedded system architecture:** Embedded systems overview, scheduling algorithms, basic concepts, first come, first served scheduling algorithm, short job first scheduling algorithm, multiprocessors, priority scheduling algorithm, ASAP and ALAP scheduling algorithm. Memory technology, mobile embedded systems, messaging and communication mechanisms.

**Data Storage and SQLite Operations:** Local data, internal and external storage, Sqlite database, table structure, CRUD operations, usage of SQLite Techniques, content provider.

**Unit IV**

**Mobile optimization by Dynamic Programming:**  Introduction of heterogeneous embedded systems and dynamic programming, Fixed time model, Probabilistic time model, Non deterministic polynomial time problems, cook’s theorem,

**Mobile optimization by Loop Scheduling:**  Introduction, basic graph models and techniques, fundamental timing optimizations, Time and power optimizations with loop scheduling, probabilistic data flow graph, loop scheduling and parallel computing.

**Unit V**

**Mobile Cloud Computing in Mobile Applications Deployment:** Introduction, Concepts of mobile cloud computing, technological structure of mobile cloud computing, difference between cloud computing and mobile cloud, mobile computing, wireless networks, cellular networks, Main techniques of mobile cloud computing, virtualization, parallel programming model, mass distributed storage, mobile cloud computing architecture.

**Efficient data synchronization on mobile devices in big data:** Overview of big data, understanding data types, categorizing big data models, current challenges in big data, Big data processing, Machine learning( Supervised, Unsupervised, Semisupervised), mobile big data storage, security and privacy issues, data de duplication.

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Student can setup Android Environment, Well understanding of Android Architecture.

CO2: Capable to customize IDE, can insert breakpoint and debugging, learn application deployment.

CO3: Learn to create different views of activity and manage layout for different activity.

CO4: Knowledge of Activity life cycle, and fragment. How intent will work for communication.

CO5: Can store Activity data on local device with database helper class.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  | L | H |  |  |  | M |  | M |  |  |  |
| CO2 |  | H |  |  | L |  |  | M |  |  |  | L |
| CO3 | H |  |  |  | H | M |  |  | L |  |  |  |
| CO4 |  |  | H | H |  |  | L | M |  |  |  |  |
| CO5 |  | L |  |  |  |  |  |  | H |  | M |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Building Android application with HTML, CSS, and JavaScript 2nd Edition O’Relly’s Publication
2. Beginning Android Programming with Android Studio

**Reference Books**

1. Android Application Development Cookbook.
2. Android 4.2 App Development Essentials
3. Android Application Development for Dummies

**Semester – V**

**Course Name: Mobile Application Development Lab**

**Course Code: BCA232**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objectives:**

CO1. Explain Android Platform, its architecture and features.

CO2. Design and implementation of user interface, database application and content providers.

CO3. Analysis of hardware components and security issues in Android.

CO4. Creating and implementing test cases to analyze performance of Android application.

CO5. Evaluating multimedia, camera and location based services in Android application.

Introduction: About Android, Pre-requisites to learn Android, Dalvik Virtual Machine & .apk file extension, Android API levels (versions & version names)

Android Java Basics: Getting started with Android development, project folder structure, simple programming, running project, generating build/APK of the app from Android Studio

First application: Creating Android Project, Android Virtual Device Creation, Set up debugging environment, Workspace set up for development, Launching emulator, debugging on mobile devices.

Basic UI design: Basics about Views, Layouts, Drawable Resources, Input controls, Input Events, Toasts.

More UI Components: Layouts - GridView and ListView, Action bar, Adapters, Menus: Option menu, context menu, sub menu, Pickers - Date and Time, Spinners.

Activity and Fragment: Activity, Fragment, Activity Lifecycle and Fragment Lifecycle.

Intents: Implicit Intents, Explicit intents, communicating data among Activities.

Navigation Drawer: Panel that displays the app’s main navigation screens on the left edge of the screen

Android Notifications – Toast, Dialogs (TimePicker, DatePicker, Progress, Alert), Notification Manager and Push Notification

Introducing SQLite - SQLiteOpenHelper and creating a database - Opening and closing a database, Working with cursors Inserts, updates, and deletes

As a term project students should implement a mobile app with the following:

* Understand the app idea and design user interface/wireframes of mobile app
* Set up the mobile app development environment

**Course Outcomes (COs): Upon successful completion of this subject students should be able to**

CO1.Apply essential Android Programming concepts.

CO2 Develop various Android applications related to layouts & rich uses interactive interfaces

CO3) Develop Android applications related to mobile related server-less database like SQLITE.

[TEXT BOOKS / REFERENCES](https://www.amrita.edu/course/mobile-application-development-lab#collapse2)

* Head first Android Development.
* Android Programming: Pushing the Limits, Wiley By Erik Hellman
* Android Application Development Black Book, Dreamtech Press, Pradeep Kothari, KLSI

**Semester – V**

**Course Name: Cryptography Lab**

**Course Code: BCA 233**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objectives:**

CO1.

CO2.

CO3.

**Experiment 1.** [Breaking the Shift Cipher](http://cse29-iiith.vlabs.ac.in/exp1/Introduction.html?domain=Computer%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Science&lab=Cryptography%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Lab)

**Experiment 2.** [Breaking the Mono-alphabetic Substitution Cipher](http://cse29-iiith.vlabs.ac.in/exp2/Introduction.html?domain=Computer%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Science&lab=Cryptography%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Lab)

**Experiment 3.** [One-Time Pad and Perfect Secrecy](http://cse29-iiith.vlabs.ac.in/exp3/Introduction.html?domain=Computer%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Science&lab=Cryptography%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Lab)

**Experiment 4.** [Message Authentication Codes](http://cse29-iiith.vlabs.ac.in/exp4/Introduction.html?domain=Computer%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Science&lab=Cryptography%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Lab)

**Experiment 5.** [Cryptographic Hash Functions and Applications](http://cse29-iiith.vlabs.ac.in/exp5/Introduction.html?domain=Computer%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Science&lab=Cryptography%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Lab)

**Experiment 6.** [Symmetric Key Encryption Standards (DES)](http://cse29-iiith.vlabs.ac.in/exp6/Introduction.html?domain=Computer%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Science&lab=Cryptography%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Lab)

**Experiment 7 .** [Symmetric Key Encryption Standards (AES)](http://cse29-iiith.vlabs.ac.in/exp7/Introduction.html?domain=Computer%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Science&lab=Cryptography%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Lab)

**Experiment 8.** [Diffie-Hellman Key Establishment](http://cse29-iiith.vlabs.ac.in/exp8/Introduction.html?domain=Computer%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Science&lab=Cryptography%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Lab)

**Experiment 9.** [Public-Key Cryptosystems (PKCSv1.5)](http://cse29-iiith.vlabs.ac.in/exp9/Introduction.html?domain=Computer%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Science&lab=Cryptography%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Lab)

**Experiment 10.** [Digital Signatures](http://cse29-iiith.vlabs.ac.in/exp10/Introduction.html?domain=Computer%20Science&lab=Cryptography%20Lab)

**Course Outcome CO: after completing the laboratory students will be capable of:**

CO 1 Implementing the cryptographic algorithms using the language they have studied

CO2 Demonstrate the practical importance of Information Security

CO 3 Analyze the implementations for time required to generate keys and encryption/decryption process also various possible attacks

CO4 Installing and configuring the proxy server and IDS

CO 5. Compare the performance of various security algorithms

**Semester – V**

**Course Name: UNIX and Shell Programming Lab**

**Course Code: BCA234**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 0 | 0 | 2 | 1 |

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1. Demonstrate how to use the following Bourne Shell commands: cat, grep, ls, more, ps, chmod, finger, ftp, etc.
2. Use the following Bourne Shell constructs: test, if then, if then else, if thenelif, for, while, until, and case.
3. Copy, move, and delete files and directories.
4. Write moderately complex Shell scripts.
5. Make a Shell script executable.
6. Create a ".profile" script to customize the user environment.
7. Execute programs written in C under UNIX environment
8. Learn tracing mechanisms (for debugging), user variables, Bourne Shell variables, read-only variables, positional parameters, reading input to a Bourne Shell script, command substitution, comments, and exporting variables. In addition, test on numeric values, test on file type, and test on character strings are covered.
9. Use advanced features of File Transfer Protocol (FTP)
10. Compile source code into object and executable modules.

**Semester – V**

**Course Name: Project**

**Course Code: BCA235**

To allow students to develop their own ideas and get experienced in industrial and research projects. It provides an opportunity in solving a real life problem by applying the knowledge gained through various courses of study and an exposure on different phases of software /system development life cycle.

**Semester – V**

**Course Name: AWS Academy Cloud Developing**

**Course Code: AWS 005A**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 0 | 0 | 3 |

**Course Objectives**

CO1 Recall cloud computing services and models.

CO2. Describe developing on AWS.

CO3. Configure AWS Identity and Access Management for programmatic access

CO4. Configure storage with Amazon S3 programmatically.

CO5. Develop with DynamoDB and explain caching.

**Module 1 – Welcome to Academy Cloud Developing (ACD)**

Cloud prerequisites objectives and overview, AWS training portal, AWS free tier, AWS educate, AWS documentation scavenger hunt.

**Module 2 – Introduction to Developing on AWS**

System development lifecycle, steps to get started developing on AWS, working with AWS SDKs, errors and exceptions, introduction to AWS X-rays, introduction to amazon cloudwatch and AWS cloudtrail.

**Module 3 – Introduction to AWS Identity and Access Management (IAM)**

Shared responsibility model , overview of IAM, authentication with IAM , authorization with IAM.

**Module 4 – Developing Storage Solutions with Amazon S3**

Introduction to Amazon S3, Creating amazon S3 buckets, working with amazon S3 objects, protecting data and managing access to amazon S3 resources.

**Module 5 – Developing NoSQL Solutions with Amazon DynamoDB**

Introduction to amazon dynamoDB, amazon dynamo DB Key concepts, partition and data distribution, secondary indexes, read/write throughput, streams and global tables, backup and restore, basic operations for amazon dynamoDB tables.

**Module 6 - Caching Information for Scalability**

Caching overview, caching with amazon cloudfront, caching with amazon elasticache, caching strategies.

**Module 7 - Introduction to Containers**

Introduction to containers, Container VS Hardware virtualization, Microservices-use case for containers, Amazon container services.

**Module 8 - Developing Solutions with Amazon SQS and Amazon SNS**

Introduction to message queues, introduction to amazon SQS, amazon SQS developer concepts, introduction to amazon SNS, amazon SNS developer concepts, introduction to amazon MQ.

**Module 9 - Developing Event-Driven Solutions with AWS Lambda**

Introduction to server less computing with AWS lambda, overview of AWS lambda, Execution model for invoking lambda functions, AWS lambda permissions, overview of authoring and configuring lambda functions, overview of deploying lambda functions.

**Module 10 - Developing Solutions with Amazon API Gateway**

Application Programming Interfaces, Amazon API Gateway, Creating a RESTful API, Controlling Access to a RESTful API, Testing a RESTful API, Deploying a RESTful API, Invoking a RESTful API, Monitoring a RESTful API.

**Module 11 - Developing Solutions with AWS Step Functions**

Workflow Coordination in Distributed Applications, Introduction to AWS Step Functions, State Types, AWS Step Functions Use Case, AWS Step Functions API.

**Module 12 - Developing Secure Applications on AWS**

Secure Network Connections. Manage Application Secrets, Authenticate with AWS Security Token Service, Authenticate with Amazon Cognito.

**Module 13 - Deploying Applications on AWS.**

Introducing DevOps, Using AWS Code Services for CI/CD, Introducing Deployment and Testing Strategies, Deploying Applications with AWS Elastic Beanstalk, Deploying Applications with AWS CloudFormation, Deploying Serverless Applications with AWS SAM.

**List of courses for Electives**

**Electives-(202)**

**Course Name: Programming in ASP.Net**

**Course Code: BCA237**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objectives:**

1. To introduce the concept of to the .NET framework, .NET Interoperation services and Different projects.
2. To demonstrate Client side programming: HTTP, CGI, Cookies, JavaScript, HTML, XML.
3. To demonstrate Server side programming: Web Forms, ASP.NET Web Services, ADO.NET Data Access.
4. To be able to create Consistent Page Layout with Master Pages.
5. To measure the User Controls, Validating User Input, Understanding Request Validation

**Syllabus**

**Unit I**

**An Introduction to ASP .NET ;-** A Tour of the IDE, The Sample Application, Different Project Types, Creating a New ASP.NET 4 Web Site, Working with Files in Your Web Site, Working with Web Forms,

**Unit II**

**Designing Your Web Pages:-** Problems of HT ML Formatting, An Introduction to CSS, Working with CSS in Visual Web Developer, Working with ASP.NET Server Controls:- Introduction to Server Controls, Ty pes of Controls, The ASP .NET State Engine

**Unit III**

**Creating Consistent Looking Web Sites:-** Consistent Page Layout with Master Pages, Using a Centralized Base Page, Themes, Skins. Navigation: - Understanding Absolute and Relative URL s, Using the Navigation Controls, Programmatic Redirection.

**Unit IV**

**User Controls:** Validating User Input, Understanding Request Validation, Processing Data at the Server, Sending E‑mail from Your Web Site, Reading from Text Files

**Unit V**

**Introducing Databases** ADO .NET Object Model, Using SQL to Work with Database Data, Creating Your Own Tables, Data Controls:- Data-bound Controls, Data Source Controls, Data Source and Data-bound Controls Working Together, Storing Your Connection Strings in web.config, AJAX, LINQ

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Web applications development using ASP.NET framework is the main outcome of this course.

CO2: Demonstrate Client side programming: HTTP, CGI, Cookies, JavaScript, HTML, XML.

CO3: Demonstrate Server side programming: Web Forms, ASP.NET Web Services, ADO.NET Data Access.

CO4: Able to create Consistent Page Layout with Master Pages.

CO5: Measure the User Controls, Validating User Input, Understanding Request Validation

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  | H |  |  |  |  | M |  |  |  |
| CO2 |  | M |  | M |  |  |  |  |  | M |  |  |
| CO3 | M |  |  |  | M |  | L |  |  |  |  | M |
| CO4 |  |  | L |  |  |  |  | M |  |  |  |  |
| CO5 | H |  |  |  | L |  |  |  | M |  |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. ASP.NET 4.5 Web Programming with C# 2012 by Mary Delamater and Anne Boehm, published by Murach, 2013**.**
2. Debugging ASP.NET (New Riders) - Jonathon Goodyear, Brian Peek, Brad Fox

**Reference Books**

1. Designing Microsoft ASP.NET Applications (Microsoft Press) - Jonathon Goodyear, Brian Peek, Brad Fox
2. Microsoft ASP.NET Step by Step (Microsoft Press) - G. Andrew Duthrie
3. Deploying and Managing Microsoft .NET Web Farms (Sams) - Barry Bloom

**Course Name: Programming in R**

**Course Code: BCA238**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objectives:**

1. To explain the key differences between the tasks of classification, clustering, regression, and dimensionality reduction
2. To identify the key differences between supervised and unsupervised learning paradigms
3. To explain how noisy observations affect the result of data mining methods.
4. To deal with missing data and Manipulate strings in R
5. To understand basic regular expressions in R and base R graphics

**Syllabus**

**Unit I**

**Introduction to R**: Basics of R, R-Environment Setup, Installation of R, Rstudio, Installing and Configuring, RStudio in Windows, Installing and Configuring, RStudio in Linux, Programming with R, Basic Data Types, Vectors, Matrices, Arrays.

Factors and data Frames: Factor Levels, Data Frame, Creating a Data Frame, Sub setting of Data Frames, Extending Data Frames, Sorting Data Frames

**Unit II**

**List:** Creating a List, Creating a Named List, Lists Operations , Conditionals and Control Flow, Relational Operators, Relational Operators and Vectors, Logical Operators, AND Operator, OR Operator, NOT Operator, Logical Operators and Vectors, Conditional Statements

**Unit III**

**Iterative Programming & Function in R**: Iterative Programming : While Loop, For Loop, Looping Over List, Loops for Vectors, Loops for Matrices, Loops for Data Frames, Loops for Lists, Functions in r

Functions: Writing a Function in R, Nested Functions, Function Scoping, Function Environment, Function Scope, Default Values for Arguments, Returning Complex, Recursion, Loading an R Package, Methods of Loading, Mathematical Functions in R, Cumulative Sums and Products, Calculus in R, Input and Output Operations

**Unit IV**

**Apply Family in R , Charts & Graphs**: Apply Family : Using apply in R, Using lapply in R, Using sapply in R, Using tapply in R, Using mapply in R.

Charts & Graphs: Pie Chart, Bar Chart, Box Plot, Histogram, Line Graph, Scatter

**Unit V**

**Data Interfaces**: Introduction to Data interfaces, CSV Files, Excel Files, Binary Files , XML files, JSON files, Web Data, , Databases

**Course Outcomes (Cos):**

CO1: Students will able to explain Basics of R programming, Installation of R, Rstudio, Installing and Configuring, RStudio in Windows.

CO2: Students will able to estimate the effects data interfaces, Conditionals and Control Flow, Relational Operators and condition flow.

CO3: Able to design data mining experiments using R and existing data mining tools.

CO4: Students will be able to learn about data interfaces, CSV files, Excel files and XML files.

CO5: Able explain the working of lists in R and will be able working with bar and charts.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  | H |  |  |  |  | M |  |  |  |
| CO2 |  | M |  | M |  |  |  |  |  | M |  |  |
| CO3 | M |  |  |  | M |  | L |  |  |  |  | M |
| CO4 |  |  | L |  |  |  |  | M |  |  |  |  |
| CO5 | H |  |  |  | L |  |  |  | M |  |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Statistical Programming in R (Oxford) Srinivasa, Siddesh, Shetty and Sowmya, June 2017.
2. Lawrence Leemis. Learning Base R. Lightning Source, 2016

**Reference Books**

1. VikramDayal. An Introduction to R for Quantitative Economics: Graphing, Simulating and Computing. Springer, 2015
2. Matthias Kohl. Introduction to statistical data analysis with R. bookboon.com, London, 2015.
3. Matthias Kohl. Introduction to statistical data analysis with R. bookboon.com, London, 2015.

**Course Name: Programming in C#**

**Course Code: BCA239**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objective:**

1. To create, compile and run object-oriented C# programs using Visual Studio.
2. To write and understand C# language constructs, syntax and semantics.
3. To develop reusable .NET components via interface realization and standard design patterns.
4. To leverage the major namespaces and classes of the .NET Framework.
5. To access databases using Language Integrated Query (LINQ).

**Syllabus**

**Unit I**

**Introduction To C#:** Introducing C#, Understanding .NET, overview of C#, Literals, Variables, Data Types, Operators, checked and unchecked operators, Expressions, Branching, Looping, Methods, implicit and explicit casting, Constant, Arrays, Array Class, Array List, String, String Builder, Structure, Enumerations, Boxing and Un- boxing.

**Unit II**

**Object Oriented Aspects Of C#:** Class, Objects, Constructors and its types, inheritance, properties, indexers, index overloading, polymorphism, sealed class and methods, interface, abstract class, abstract and interface, operator overloading, Delegates, Events, Errors and Exception, Threading.

**Unit III**

**Application Development On .Net:** Building windows application, Creating our own window forms with events and controls, menu creation, inheriting window forms, SDI and MDI application, Dialog Box(Modal and Modeless), accessing data with ADO.NET, Dataset,DataAdapter, updating database using stored procedures, SQL Server with ADO.NET, Handling Exceptions, Validating Controls, Windows Application Configuration.

**Unit IV**

**Web Based Application Development On .Net:** Programming web application with web forms, ASP.NET introduction, working with XML and .NET, Creating Virtual Directory and Web Application, session management techniques, Web.Config, Web Services, passing datasets, returning datasets from web services, handling transaction, handling exceptions, returning exceptions from SQL Server.

**Unit V**

**Clr And .Net Framework:** Assemblies, Versoning, Attributes, Reflection, Viewing Meta Data, Type Discovery, Reflection on type, Marshalling, Remoting, Security .

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Create, compile and run object-oriented C# programs using Visual Studio.

CO2: Write and understand C# language constructs, syntax and semantics with object oriented paradigms.

CO3: Develop .NET application development through SDI, MDI and ADO.

CO4: Develop web based applications with the help of .net framework.

CO5: Access databases using CLR and .net framework.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  | H |  |  |  |  |  |  |  |  |  |  |
| CO2 |  |  | H |  |  |  |  |  | M |  |  |  |
| CO3 |  |  |  |  | H |  |  |  |  |  |  | M |
| CO4 | M |  |  |  |  |  | H |  |  |  | L |  |
| CO5 |  | H |  |  |  |  |  | M |  |  |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Programming in C#, Oxford, Harsh Bhasin, March 2014.
2. “C# 4.0 The Complete Reference” by Herbert Schildt

**Reference Books**

1. “Beginning Visual C# 2010” by Karli Watson
2. “Essential C# 4.0” by Mark Michaelis and Eric Lippert
3. “Programming in C#” by Balagurusamy

**Course Name: Professional Java**

**Course Code: BCA240**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objectives::**

1. To provide thorough understanding of each topic through extensive examples along with the program codes and screenshots
2. To be extensive coverage of advanced topics on Java from tools to enterprise Java
3. To provide ample application-based examples, with step-by-step explanations
4. To provide relevant software installation and configuration information wherever necessary
5. To comprises keywords, objective-type questions (with answers) and subjective-type questions for students at the end of all the chapters

**Syllabus**

**Unit I**

**Java Naming and Directory Interface:** Naming Concepts, Directory Concepts,Java Naming and Directory Interface, An Example, Specifying JNDI Properties, Name Servers, Using Apache DS, Calculator RMI Application Using JNDI, Naming Operations, Working with Directory.

**Unit II**

**Java Message Service:** Messaging, JMS API, JMS Components, Messaging Models, Programming Model, Installing Open MQ, Writing JMS Application, Writing a P2P Producer, Writing a P2P Synchronous Consumer, Writing a P2P Asynchronous Consumer,Writing a Pub/Sub producer, Writing a Pub/Sub Synchronous Consumer, Running this Example, Writing a Pub/Sub Asynchronous Consumer, Browsing Queue, Using JNDI, Reliability Mechanisms, Transacted Session.

**Unit III**

**Introduction to J2EE:** Overview of J2EE, Introduction to JavaBeans**,** Bean Builder, Advantages of JavaBeans, BDK Introspection, Properties, BeanInfo Interface, Persistence, Customizer, JavaBeans API, EJB, Introduction to Struts Framework.

**Unit IV**

**Java and CORBA:** Introduction, CORBA Architecture, Java IDL, Developing CORBA Applications, Compiling Applications, Running the Application, Using Properties, Stringified Object Reference, Using URL, Using Tie, Persistent Objects, Callback, Using Dynamic Invocation Interface, Using Dynamic Skeleton Interface, Using Out and In out Parameter, RMI-IIOP, IDL to Java Language Mapping.

**Unit V**

**Java Server Faces:** Introduction, First Application, Request Processing Life Cycle, Tracing Phases, Managed Bean, Accessing Managed Bean Programmatically, Basic JSF Tags, Expression Language, AJAX, Event Handling, An Event Handling Example, Page Navigation.

**Course Outcomes (Cos):**

**Upon successful completion of this subject students should be able to:**

CO1: Revise object oriented features of java language and develop java applet programming using various techniques.

CO2: Handling exceptions and develop multi- threaded applications.

CO3: Develop applications using collection framework and concepts of generic programming.

CO4: Develop applications using Abstract Window Toolkit

CO5: Develop server side programs using Servlets and develop Java Server Pages applications using JSP Tags.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  | H | H |  |  |  |  |  | M |  |  |  |
| CO2 |  |  | M |  |  | L |  |  |  | M |  |  |
| CO3 |  |  |  | H |  |  |  |  |  |  |  | M |
| CO4 | L |  |  |  |  |  |  | H |  |  |  |  |
| CO5 |  | M |  |  | H |  |  |  |  | L |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Programming with Java A Primer, E.Balaguruswamy Tata McGraw Hill Companies
2. Advance Java Programming, Oxford, Uttam Kumar Roy, April 2015.

**Reference Books**

1. Java Programming Language Ken Arnold Pearson
2. The complete reference JAVA2, Herbert schildt. TMH

**Course Name: Programming in ASP.Net Lab**

**Course Code: BCA241**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1. Write a program to display the following feedback form.
2. The different options for the list box must be ASP-XML, DotNET, JavaPro and UNIX, C, C++. When the Submit Form button is clicked after entering the data, a message as seen in the last line.
3. Write a program to display three images in a line. When any one of the images is clicked, it must be displayed below. On clicking the displayed image it must be cleared.
4. Write a simple ASP.NET program to display the following Web Controls:
5. A button with text “clicks me”. The button control must be in the center of the form.
6. A label with a text hello
7. A checkbox.
8. Write a program to display “Welcome To Radiant” in the form when the “click” button is clicked. The form title must be ASP.NET.
9. Write a program containing the following controls:
10. A ListBox
11. A Button
12. An Image
13. A Label
14. The listbox is used to list items available in a store. When the user clicks on an item in the listbox, its image is displayed in the image control. When the user clicks the button, the cost of the selected item is in the control

**Course Name: Programming in R Lab**

**Course Code: BCA242**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

Following are the list of sample programs. Students have to perform accordingly.

1. Illustrate addition, multiplication and division between vectors.
2. Enumerate multiplication and division operations between matrices and vectors in R console.
3. Write the command in R console to create a list containing a vector, a matrix and a list. Also give names to the elements in the list and display the list.
4. Write the command in R console to add a new element at the end of the list and display the same.
5. Write the command in R console to delete the fourth element from a list and display the resultant list.
6. Write the command in R console to update the third element of the list and display the resultant list.
7. Write the command in R console to create two lists, each containing 5 elements. Convert the list into vectors and perform addition on the two vectors. Display the resultant vector.
8. Write an R programme to print the values in vectors using the while loop.
9. Write an R programme to print the values in vectors using for loop.
10. Write an R programme to create four vectors namely patientid, age, diabetes and status. Put these four vectors into a Data frame patientdata and print the values using the for loop.
11. Write an R programme to print HELLO 10 times using for loop.
12. Write an R programme to print the Fibonacci series.
13. Write an R programme to print if the given number is ODD/EVEN.
14. Write an R programme to print the first 100 prime numbers.
15. Write an R programme to calculate the factorial of a number.
16. Write an R programme to calculate LCM of 2 numbers.
17. Write an R programme to calculate GCD of 2 numbers.
18. Write a programme to illustrate the use of local and global variable.
19. Write a programme to multiply two numbers using a function with a default value. Assume default value as NULL.
20. Demonstrate the creation of a complex number.
21. Add two complex numbers using the complex functions and test whether the sum of the complex numbers is complex or not.
22. Write a programme to calculate factorial of a number using recursive computation.
23. Write a programme to sum n natural numbers.
24. Write a programme to find nth Fibonacci number using recursive computation.
25. Write a programme to calculate the GCD of two numbers using recursive computation.
26. Write the commands in R console to find mean number of leaves for each day (data frame) using apply function.
27. Write the command in R console to specify the columns that needs to be excluded in the apply function.
28. Write the command in R console to change a value in the duckweed data frame.
29. Write the command in R console to determine the proportion of the total number of leaves counted on each day.
30. Write the command in R console to obtain mean using tapply function by considering a vector having 10 normal and 10 uniform variables. Assume that these vectors have three groups.
31. Calculate the mean of ozone, solar radiation, and wind within each month using lapply for air quality dataframe.
32. Calculate the mean of ozone, solar radiation, and wind within each month using sapply for air quality dataframe.
33. Using R pie chart, demonstrate the percentage conveyance of various ways for travelling to office such as walking, car, bus, cycle and train.
34. Using a chart legend, Show the percentage conveyance of various ways for travelling to office such as walking, car, bus, cycle and train.
35. Using R bar chart, demonstrate the percentage conveyance of various ways for travelling to office such as walking, car, bus, cycle and train.
36. Using box plots demonstrate the relation between the cars speed and the distance taken to stop.
37. Using R histogram, demonstrate the relation between the cars speed and the distance taken to stop.
38. Using R line graphs, demonstrate the relation between the cars speed and the distance taken to stop.
39. Using scatters plots in R, demonstrate the relation between the cars speed and the distance taken to stop.
40. Write the commands in R console to read a CSV file and display the number of rows and columns.

**Course Name: Programming in C# Lab**

**Course Code: BCA243**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Simple Programs**

1. Write a Program in C# to enter two distinct numbers and find the greater amongst the two.
2. Write a Program in C# to find the roots of a Quadratic Equation.
3. Write a Program in C# to enter three sides of a triangle and find the type of that triangle. (i.e. equilateral, isosceles, or scalene)
4. Write a Program in C# using switch statement to assign grades to the students on the basis of the marks obtained. The policy for assigning the grades is given in the below table:
5. Write a menu based program in C# using do-while and switch statement to repeatedly perform addition, subtraction, and multiplication of two numbers until the user wants to quit.
6. Write a Program in C# to demonstrate the Stack class in the System. Collections namespace.
7. Write a Program in C# to swap two numbers using ref keyword.
8. Write a Program in C# to demonstrate the concept of delegate.
9. Write a program in C# to convert the string into a character array using the ToCharArray() method of String class.
10. Write a program in C# to enter the elements of a 3 x 3 matrix and then display the matrix.
11. Write a program that asks the user to enter the details of a student s1, which is an instance of a class Students. The Student class has name:string, age:int, and fees:double as its attributes. The input and output should be done via functions getdata() and putdata() respectively.
12. Write a program to illustrate the concept of property in C#.
13. Write a program to demonstrate abstract class and abstract methods in C#.
14. Write a program in C# to build a class which implements an interface.
15. Implement Queue in C# using the existing System.Collections.Generic namespace.
16. Program to add 5 numbers using Console Application. Numbers will be passed in Main() method as Command Line Arguments.
17. Create your own DLL (dynamic link library) and use it in a C# program.
18. Create a Windows form application to add, and subtract two numbers.
19. Create a Windows form application. It shows a simple Login screen, which is accessible by the user. The user will enter the required credentials and then will click the Login button to proceed.
20. Develop a MDI & Database based small project in C# comprising of 5-6 Windows Forms.

**Course Name: Professional Java Lab**

**Course Code: BCA244**

1. Write a JAVA Program to show validation of user using servlet
2. Write a JAVA Program of calling one servlet by another servlet
3. Write a JAVA Program to show validation of user using jsp
4. Write a JAVA Program to display message on browser using jsp
5. Write a JAVA Program to demonstrate use of beans
6. Write a JAVA Program to set scope of beans
7. Write a JAVA Program to demonstrate working of rmi
8. Write a JAVA Program to create simple web service.
9. Write a JAVA Program to use <jsp:useBean> Tag in JSP
10. Write a JAVA Program for session management in JSP
11. Write a JAVA Program to implement the URL.
12. Write a JAVA Program to implement the InetAddress.
13. Write a JAVA Program for Sending e-mail in Java.
14. Write a JAVA Program to implement Single Client-Server Communication.
15. Write a JAVA Program to implement the List.
16. Write a JAVA Program to implement the JTrees.
17. Write a JAVA Program to implement the JTable.
18. Write a JAVA Program to implement Remote Method Invocation.
19. Write a JAVA Program to implement Corba.
20. Write a JAVA Program to implement JNDI.

**Electives-(300)**

**Course Name: Data Warehousing and Data Mining**

**Course Code: BCA250**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
| 3 | 1 | 0 | 4 |

**Course Objective:**

1. To know the basic concept of data mining, Architecture of a Data Mining System and the Knowledge Discovery Process
2. To provide the fundamental concepts of data warehousing technology.
3. To incorporates a step-by-step approach to designing and building a data warehouse.
4. To provides numerous review questions, multiple choice questions and other exercises at the end of each chapter.
5. To contain a running fabricate case-study to bring out practical aspects of building a data warehouse.

**Syllabus**

**Unit I**

**Introduction to Data mining**: Introduction, Architecture of a Data Mining System, The Knowledge Discovery Process , Integrating Data Mining with Data Warehouse, Related Areas of Data Mining , Data Mining Techniques

**Unit II**

**Data Warehousing**: Introduction to Data Warehousing: Historical Background, Increasing Demand for Strategic Information, Data Warehouse Defined, Data Warehouse Users, Benefits of Data Warehousing, Concerns in Data Warehousing. Features of a Data Warehouse, Data Granularity, the Information Flow Mechanism, Metadata, Two Classes of Data, The Lifecycle of Data, Data Flow from Warehouse to Operational Systems.

**Unit III**

**Architecture, Planning and Project Management**: Architecture, Characteristics of Data Warehouse Architecture, Data Warehouse Architecture Goals, Data Warehouse Architecture, Data Warehouse and Data Mart, Building Data Marts and its issues, Building Data Marts, Other Data Mart Issues.

Planning and Project Management: Project Management Principle, Data Warehouse Readiness Assessment, Data Warehouse Project Team, Planning for the Data Warehouse, Data Warehouse Project Plan, Planning for the Data Warehouse Server, Capacity Planning, Selecting the Operating System, Selecting the Database Software.

**Unit IV**

**Schema & Dimensional Modeling**: Schema: Dimensional Modeling, The Star Schema, The Snowflake Schema, Aggregate Tables, Fact Constellation Schema, The Strengths of Dimensional Modeling, Data Warehouse and the Data Model.

Dimensional Modeling: Characteristics of a Dimension Table, Characteristics of a Fact Table, T Types of Dimension Tables, Keys in the Data Warehouse (Star) Schema, Enhancing the Data Warehouse Performance, Technology Requirements.

**Unit V**

**Testing, Growth, Maintenance & OLAP**: Data Warehouse Design Review, Developing the Data Warehouse Iteratively, Testing, Monitoring the Data Warehouse, Tuning the Data Warehouse. OLAP: Need For OLAP, OLAP and Multidimensional Analysis, OLAP Functions, OLAP Applications, OLAP Models, OLAP Design Considerations, OLAP Tools and Products, Existing OLAP Tools, Data Design, Administration and Performance, OLAP Platforms.

**Course Outcomes (COs):**

**On successful completion of this course, the learner will be able to**

CO1: Describe the fundamental concepts, benefits and problem areas associated with data warehousing.

CO2: Describe the various architectures and main components of a data warehouse.

CO3: Design a data warehouse, and be able to address issues that arise when implementing a data warehouse.

CO4: Understand various schemes and data models.

CO5: Compare and contrast OLAP and data mining as techniques for extracting knowledge from a data warehouse.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | H |  | H |  |  |  |  | H | M |  |  |  |
| CO2 |  |  |  | H |  | H |  |  |  | L |  |  |
| CO3 |  |  |  |  |  |  |  |  |  |  |  | M |
| CO4 |  |  |  | M |  |  |  | H |  |  |  |  |
| CO5 |  | H |  |  |  |  | M |  | H |  | M |  |

H = Highly Related; M = Medium; L = Low

**Text Books:**

1. Jiawei Han and MichelineKamber, “Data Mining- Concepts and Techniques”, (3e), Morgan Kaufmann Publishers, 2011
2. VikramPudi, ''Data Mining" , Third Edition, Oxford university Press

**Reference Books :**

1. PaulrajPonniah, “Data Warehousing”, (2e), Wiley India Pvt. Ltd., 2010
2. H. Witten and E. Frank, “Data Mining: Practical Machine Learning Tools and Techniques”, Morgan Kaufmann, 2000.
3. M.H.Dunham, “Data Mining: Introductory and Advanced Topics”, Pearson Education 2006.

**Course Name: Big Data Analysis**

**Course Code: BCA252**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objectives:**

1. To know the fundamental concepts of big data and analytics.
2. To explore tools and practices for working with big data
3. To learn about stream computing.
4. To know about the research that requires the integration of large amounts of data.
5. To know about clustering and classification.

**Syllabus**

**Unit I**

**Introduction To Big Data:** Evolution of Big data - Best Practices for Big data Analytics - Big data characteristics - Validating- The Promotion of the Value of Big Data - Big Data Use Cases- Characteristics of Big Data Applications - Perception and Quantification of Value -Understanding Big Data Storage – A General Overview of High-Performance Architecture - HDFS – Map Reduce and YARN – Map Reduce Programming Model

**Unit II**

**Clustering and Classification:** Advanced Analytical Theory and Methods: Overview of Clustering - K-means - Use Cases -Overview of the Method - Determining the Number of Clusters - Diagnostics - Reasons toChoose and Cautions .- Classification: Decision Trees - Overview of a Decision Tree – TheGeneral Algorithm - Decision Tree Algorithms - Evaluating a Decision Tree - Decision Trees inR - Naïve Bayes - Bayes‘ Theorem - Naïve Bayes Classifier.

**Unit III**

**Association and Recommendation System:** Advanced Analytical Theory and Methods: Association Rules - Overview - Apriori Algorithm Evaluation of Candidate Rules - Applications of Association Rules - Finding Association& findingsimilarity - Recommendation System: Collaborative Recommendation- Content BasedRecommendation - Knowledge Based Recommendation- Hybrid Recommendation Approaches.

**Unit IV**

**Graph Memory:** Using Graph Analytics for Big Data: Graph Analytics - The Graph Model - Representation asTriples - Graphs and Network Organization - Choosing Graph Analytics - Graph Analytics UseCases - Graph Analytics Algorithms and Solution Approaches - Technical Complexity ofAnalyzing Graphs- Features of a Graph Analytics Platform - Considerations: DedicatedAppliances for Graph.

**Unit V**

**Stream Memory:**Introduction to Streams Concepts – Stream Data Model and Architecture -Stream Computing, Sampling Data in a Stream – Filtering Streams – Counting Distinct Elementsin a Stream – Estimating moments – Counting oneness in a Window – Decaying Window – Realtime Analytics Platform(RTAP) applications - Case Studies - Real Time Sentiment Analysis,Stock Market Predictions.

**Course Outcomes (COs):**

**On successful completion of this course, the learner will be able to :**

CO1: Identify the difference between structured, semi-structured and unstructured data.

CO2: summarize the challenges of big data and how to deal with the same.

CO3: Explain the significance of NoSQL databases.

CO4: Explain about Hadoop Ecosystem.

CO5: Identify the difference between Pig and Hive.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  | M |  |  |  | M |  |  |  | M |  |  |
| CO2 |  |  |  | M |  |  |  | M |  |  |  |  |
| CO3 |  |  |  |  | M |  |  |  |  |  | M |  |
| CO4 |  |  |  |  |  |  | H |  |  |  |  |  |
| CO5 | H | M |  |  |  | M |  |  | M |  |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. AnandRajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", CambridgeUniversity Press, 2012.
2. David Loshin, "Big Data Analytics: From Strategic Planning to Enterprise Integration

**Reference Books**

1. Tom White, “Hadoop: The Definitive Guide”, Second Edition, O’Reilly Yahoo Press.
2. Robert D. Schneider, “Hadoop for Dummies”, Wiley.
3. VigneshPrajapati, “Big Data Analytics with R and Hadoop”, Packt Publishing.

**Course Name: Introduction to Salesforce**

**Course Code: BCA254**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objective:**

1. To introduce cloud computing and sales force approaches.
2. To understand the concept and features of CRM, Data Migration Import Wizard and Data Loader.
3. To describe needs of APEX as a Programming Language, Various Statements and Operators.
4. To demonstrate class and visual force.
5. To be able to understand the concept of Trigger Context Variables, Validation and Automation, Components, Client Controller and Server Controller.

**Syllabus**

**Unit I**

**Introduction to Cloud Computing & Sales force:** On premise and Cloud approach, IaaS, PaaS and SaaS, Clouds, Versions, Editions & Environments, Product & Project Development

MVC Pattern and Multitenent Architecture: Configuration, Development, Testing and Admin, along with 15 steps of Salesfroce.com, MVC Diagram, Multi-Tenant Diagram

Starting Designing Applications on Force.com: Basic Vocabulary, Registration, Navigation and Principles, Understanding the Same with Lightning, Relationships, All 6 Types, Validation, Page Layouts and Record Types

**Unit II**

**CRM:** Campaign, Lead, Account, Contact and Opportunity, Campaign, Lead, Account, Contact and Opportunity, Product, Proce Book, Quote, Forecast, Order, Asset, Case, Solution, Activity, Task, Event, Calendar, PaaS features for CRM.

Data Migration Import Wizard and Data Loader: Reports and Dash-Boards, Work flows, Process Builder, Approval Process, Security Profile, Role, Queues, Public Group, OWD, Sharing Rules, Permission Sets.

**Unit III**

**Introduction to Programming:** Need of APEX as a Programming Language, Various Statements and Operators, 9 types of operators and 5 types of statements, System.debug, Comments, Indentation, Naming Conventions, OOP, Business Class , Test Class, Test Methods, Creating object, calling methods, executing a program, Constructor / Method / Operator Over Loading / this, Default , Parameterised .

**Unit IV**

**Class and Visual Force:** Static and non-static members of a class, Class Summary, try catch, Collections list: Set, Map, Exception Handling, Assertions and Annotations: 6 types of Assertions and 3 Annotations, Database Statements: DML , MC inter relation, Database Statements : SOQL / SOSL, Visual Force - Types of Tags: 6 Types, Visual Force -Static / Dynamic Data, Org Sobject $User: {! } $User, Standard Controllers: Save, Save and New, Cancel ,Edit, Delete, Explanation of for each loop / Standard list Controller : Previous, Next, First Last, Custom Controller, VC / MVC Examples: Add numbers, Insert and fetch Record

**Unit V**

**Custom List Controller:** Insert and fetch Records, Schedule Apex: Interface, Batch Apex: 50 K, 50 Million, Triggers: if and its types, Trigger Context Variables, Validation and Automation, Components, Client Controller, Server Controller

**Course Outcomes (COs):**

**Upon successful completion of this subject students will able**

CO1: Introduce cloud computing and sales force approaches.

CO2: Able to understand the concept and features of CRM, Data Migration Import Wizard and Data Loader.

CO3: Describe needs of APEX as a Programming Language, Various Statements and Operators.

CO4: To demonstrate class and visual force.

CO5: Able to understand the concept of Trigger Context Variables, Validation and Automation, Components, Client Controller and Server Controller.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  | H |  |  |  | L |  |  |  |  | H |  |
| CO2 |  |  |  | H |  |  |  |  |  | M |  |  |
| CO3 | H |  |  |  |  |  |  | M |  |  |  | M |
| CO4 |  |  |  | H |  |  |  |  |  |  |  |  |
| CO5 |  | M |  |  | H |  |  |  | M |  |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books:**

1. Wes Nolte ,‎ Jeff Douglas , “Salesforce Handbook” (3e), Lulu.com, 2011.
2. David Taber, “Salesforce.com Secrets of Success: Best Practices for Growth and Profitability”, 1 edition, Prentice Hall, 2009.

**Reference Books:**

1. Tom Wong, Liz Kao & Matt Kaufman “Salesforce for Dummies”,4th Revised edition edition ,John Wiley & Sons, 2010
2. Visual force practices by Michael Floyd, Don Robins, Dan Appleman, et al.
3. Development with the Force.com Platform, Jason Ouellette

**Course Name: Artificial Intelligence and Expert System**

**Course Code: BCA255**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objectives:**

1. To understand AI methodology and fundamentals; intelligent agents.
2. To understand search algorithms; game playing; supervised and unsupervised learning; decision tree learning.
3. To learn neural networks, nearest neighbor methods, dimensional reduction, clustering.
4. To understand kernel machines, support vector machines; uncertainty and probability theory.
5. To understand probabilistic reasoning in AI.

**Syllabus**

**Unit I**

**History:**Artificial Intelligence: History and Applications, Intelligence, Communication, Learning, Artificial Intelligence, Progress of Artificial Intelligence, Modeling, Simulation, and AI, Intelligent Systems, Comparison of Various Intelligent Systems

**Unit II**

**Knowledge Representation:**Reasoning, Issues, and Acquisition, Propositional Calculus, Predicate Calculus, Rule-based Knowledge Representation, Symbolic Reasoning Under Uncertainty, Basic Knowledge Representation Issues, Knowledge Acquisition

**Unit III**

**Artificial Intelligence Problem-solving Languages**: PROLOG , Control Structures used in Rule-Based Systems , Search, Search Strategies Using Search Techniques , Implementation of Search, Construction and Debugging Methods for a Program, LISP, Problem-solving by LISP, Implementation of LISP , Comparison of LISP and PROLOG

**Unit IV**

**Fuzzy Systems:**Foundation of Fuzzy Systems, Fuzzy Relations ,Arithmetic Operations of Fuzzy Numbers, Linguistic Descriptions and their Analytical Forms , De- fuzzification Methods, Fuzzy Logic in Control and Decision-making Applications,

**Unit V**

**Expert Systems:** Features ,Characteristics, Architecture, Goals, Basic Activities, Advantages, Stages in the Development of an Expert System, Probability-based Expert Systems, Expert System Tools , Applications

**Course Outcomes:**

**On successful completion of this course students will be able to:**

CO1: Knowledge of what constitutes "Artificial" Intelligence and History of Artificial Intelligence.

CO2: Explain Implementations Applications related to Artificial Intelligence.

CO3: Implement Expert system technologies and Fuzzy logic theory.

CO4: Ability to apply Artificial Intelligence with neural network and genetic algorithm.

CO5: Explain the Swarm Artificial Intelligence techniques.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  | M |  |  |  | M |  |  |  | M |  |  |
| CO2 |  |  |  | M |  |  |  | M |  |  |  |  |
| CO3 |  |  |  |  | M |  |  |  |  |  | M |  |
| CO4 |  |  |  |  |  |  | H |  |  |  |  |  |
| CO5 | H | M |  |  |  | M |  |  | M |  |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. N. P. Padhy, ''Artificial Intelligence and Intelligent Systems", 1 edition, Oxford University Press , 2005.
2. Don W. Patterson,“Introduction to Artificial Intelligence and Expert Systems”, PHI Publication,2006.

**Reference Books**

1. Stuart Russell and Peter Norvig,“Artificial Intelligence A Modern Approach”, 3‘rd Edition , Pearson Education, 2010.
2. 2. Elaine Rich, Kevin Knight, Shivashankar B. Nair, “Artificial Intelligence”, 3‘rd Edition, TataMcGraw Hill Edition, 2008.

Saroj Kaushik, “Artificial Intelligence”, (1e),Cengage Learning Publications,

**Course Name: Design & Analysis of Algorithms**

**Course Code: BCA256**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objective:**

1. To design effective, efficient, elegant, and readable algorithms for various classes of computing problems
2. To learn good principles of algorithm design;
3. To determine space and time complexities of algorithms by the use various algorithm design techniques like (divide and conquer, backtracking, greedy, etc.)
4. To become familiar with fundamental data structures and with the manner in which these data structures can best be implemented; become accustomed to the description of algorithms in both functional and procedural styles;
5. To learn how to apply their theoretical knowledge in practice

**Syllabus**

**Unit I**

Introduction, algorithms specification, time and space complexity, performance analysis, recurrence relations. Divide and Conquer – finding max min.

**Unit II**

Dynamic Programming and Greedy Methods – Huffman tree construction, Knapsack problem, 0/1 Knapsack problem,least common subsequence**,** matrix chain multiplication. Backtrack: 4-queen problem, Branch and Bound: assignment problem

**Unit III**

Graph algorithms–flow problems, String Matching Algorithms: Naive algorithm, automata and KMP matcher algorithms, Boyer-Moore algorithm

**Unit IV**

Number Theory Problems – CRT, GCD algorithms, modular arithmetic, Lower Bound Theory; Approximate Algorithms – Set cover, vertex cover, .Randomized Algorithms – Las Vegas and Monte Carlo methods

**Unit V**

NP Completeness: Definitions of P, NP-Hard and NP-Complete Problems. Decision Problems.**.**

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Able to design effective, efficient, elegant, and readable algorithms.

CO2: What are the various methods of calculating complexity?

CO3: Which method is best for different algorithms?

CO4: About computational geometry and lower bound theory etc.

CO5: Able to solve the NP completeness problem.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  | H |  |  |  |  |  | M |  |  |
| CO2 |  | H |  |  | L |  |  | M |  |  |  |  |
| CO3 |  |  | H |  |  |  |  |  |  |  | M |  |
| CO4 |  |  |  |  |  |  | H |  |  |  |  |  |
| CO5 | H |  |  | L |  |  |  |  | M |  |  |  |

H = Highly Related; M = Medium; L = Low

**Textbooks**

1. Cormen, Leizerson&Rivest, Introduction to algorithms, Prentice-Hall. 2002
2. Horowitz &Sahni, Fundamentals of Computer Algorithms, Galgotia Publication. 1999

**Reference Books:**

1. Aho, HopCroft, Ullman, The Design and Analysis of Computer Algorithms, Addison-Wesley. 2001.
2. Introduction to Design and Analysis of Algorithms, Anny Levitin, Person Education Press. 2007.
3. Gilles Brassard & Paul Bratley, Fundamental Algorithms, Prentice-Hall. 1998

**Course Name: Compiler Design**

**Course Code: BCA257**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objectives:**

1. To understand, design and implement a lexical analyzer.
2. To understand, design and implement a parser.
3. To understand, design code generation schemes.
4. To understand optimization of codes and run time environment.
5. To understand various compiler writing tools.

Syllabus

Unit I

Introduction to Compiler: **Theory of Computer Languages, Design of a Language, Evolution of Compilers, Stages of Compilation: Lexical Analysis, Syntactic Analysis, Semantic Analysis, Intermediate Code Generation, Code Optimization, Code Generation, Symbol Table Management.**

**Unit II**

**Lexical Analysis and Syntax Analysis:** Lexical Analysis: Alphabets and Tokens in Computer Languages, Representation of Tokens and Regular Expression, Token Recognition and Finite State Automata, Implementation, Error Recovery.

### Unit III

1. **Syntax Analysis:** Context-free Grammar and Structure of Language, Parser and its Types, Top down Parser, Bottom-up Parser, Parser Generator Tool (Yacc).

**Unit IV**

**Code Optimization:**Need for Optimization, Objectives, Performance factors, Writing Optimized Code at User Level, Construction of Basic Blocks and Processing, Data-flow Analysis Using Flow Graph, Principal Sources of Optimization and Transformations, Alias, Procedural Optimization, Loops in Flow Graphs, Loop Optimization.

**Unit V**

**Code Generations and Compiler Writing Tools**: Issues in Code Generation, Target Machine Architecture, Subsequent Use Information,Register Allocation, Directed Acyclic Graph Representation of Basic Blocks, Code Generation From Intermediate Code, Peephole Optimization, Code Scheduling, Lexical Tools.

**Course Outcomes (COs):**

**On successful completion of this course, the learner will be able to**

CO1: Learn the design and develop a comprehensive Compiler for a given language or anIntroduction.

CO2: Analyze and able to design a compiler development with lexical and syntax analysis.

CO3: Describe all aspects of Run time storage with intermediate Code generation.

CO4: Implement various code optimization techniques.

CO5: Implement various optimization and code generation algorithms for the design of acompiler.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 |  |  |  | H |  |  |  |  |  | M |  |  |
| CO2 |  | H |  |  | L |  |  | M |  |  |  |  |
| CO3 |  |  | H |  |  |  |  |  |  |  | M |  |
| CO4 |  |  |  |  |  |  | H |  |  |  |  |  |
| CO5 | H |  |  | L |  |  |  |  | M |  |  |  |

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Compiler Design, Oxford University, Muneeswaran, November 2012.
2. “Engineering a Compiler” by Keith D Cooper and Linda Torczon

**Reference Books**

1. “Compilers: Principles, Techniques and Tools” by Alfred V Aho and Ravi Sethi
2. “Compiler Design in C” by Allen I Holob
3. “Elements of Compiler Design” by Meduna

**Course Name: Theory of Computation**

**Course Code: BCA258**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objectives:**

1. To introduce students to the mathematical foundations of computation including automata theory.
2. To be able to understand the concept of theory of formal languages and grammars and the notions of algorithm, decidability, complexity, and computability.
3. To enhance/develop students' ability to understand and conduct mathematical proofs for computation and algorithms.
4. To enhance the ability to understand the concept of Regular Expression Formalism.
5. To be able to describe formation of Turing machine by Grammars.

**Syllabus**

**Unit I**

**Introduction**: Basic Concepts, Sets, Relations, Graphs, Languages, Mathematical Induction, Finite State Machines FSM: Concept of Basic Machine, Finite State Machine, Finite Automata,

Deterministic Finite Automata, Non-deterministic Finite Automata, Equivalence of NFA and DFA, NFA with є-Transitions, Equivalence of NFA and NFA with є-Transitions, Equivalence of DFA and NFA with є-Transitions, DFA Minimization

**Unit II**

**Regular Expression:**  Formalism, Examples of Regular Expressions, Equivalence of Regular Expressions and Finite Automata, DFA to Regular Expression Conversion, Regular Sets and their Closure Properties, Pumping Lemma for Regular Languages, Decision Algorithms for Regular Sets, Applications of Regular Expressions and Finite Automata.

**Unit III**

**Turing Machine:** Elements of a Turing Machine, Turing Machine Formalism, Instantaneous Description, Transition Graph for Turing Machine, Complexity of a Turing   
Machine, Composite and Iterative Turing Machines, Universal Turing Machine, Multi-tape Turing Machine, Multi-stack Turing Machine, Multi-track Turing Machine.

**Unit IV**

**Grammar:** Constituents of Grammar, Formal Definition of Grammar, Grammar Notations, Derivation Process, Derivation Tree, Context-free Languages, Ambiguous Context-free Grammar, Simplification of Context-free Grammar, Normal Forms, Chomsky Hierarchy, Equivalence of Right-linear and Left-linear Grammars ,Equivalence of Regular Grammars and Finite Automata Pumping Lemma for Context-free Languages.

**Unit V**

**Automata:** Elements of a PDM, Pushdown Automata, Finite Automata vs PDAPDA Accepting CFLs, DPDA vs NPDA, Equivalence of CFG and PDA, Closure Properties of CFLs, Additional PDA Examples.Parsing, Bottom-up Parsing, Working of a Bottom-up Parser, Automatic Construction of Bottom-up Parser.

**Course Outcomes (COs):**

**On successful completion of this course, the learner will be able to**

CO1: Students are able to understand basic terms TOC and FSM.

CO2: Students are able to understand Regular Expression and its fundamentals.

CO3: Students will be able to describe formation of Turing machine by Grammars.

CO4: Students will be able to understand the different-2 types of grammars

CO5: Students will be able to describe the PDA.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcome** | **Program Outcome** | | | | | | | | | | | |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| CO1 | H |  | H |  |  |  |  | H | M |  |  |  |
| CO2 |  |  |  | H |  | H |  |  |  | L |  |  |
| CO3 |  |  |  |  |  |  |  |  |  |  |  | M |
| CO4 |  |  |  | M |  |  |  | H |  |  |  |  |
| CO5 |  | H |  |  |  |  | M |  | H |  | M |  |

H = Highly Related; M = Medium; L = Low

**Text Books:**

1. Vivek Kulkarni ''Theory of Computation" , Oxford University Press. 2013
2. Peter Linz, “An Introduction to Formal Languages and Automata”, (5e),Jones & Bartlett Learning, 2011

**Reference Books:**

1. John C Martin, “Introduction to Languages and the Theory of Computation”, (3e), McGraw Hill, 2007.
2. J E Hopcroft, Rajeev Motwani & Jeffrey D Ullman, “Introduction to Automata Theory, Languages and Computation”, (3e), Pearson Education, 2006.
3. K. L. P. Mishra, N. Chandrashekharan, “Theory of Computer Science”, (3e),PHI publications, 2007.

**Course Name: Robotics Process Automation**

**Course Code: BCA259**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Syllabus**

Unit-1

Programming Basics & Recap:- Programming Concepts Basics, Understanding the application, Basic Web Concepts,Protocols,EmailClients,DataStructures,DataTables,Algorithms,Software.

Processes, Software Design, SDLC:- Programming Concepts Basics – 2 Scripting,.Net Framework,.Net ,Fundamentals, XML, Control structures and functions, XML, HTML, CSS, Variables & Arguments.

RPA Basics:- History of Automation, What is RPA,RPA vs Automation, Processes & Flowcharts, Programming Constructs in RPA, What Processes can be Automated, Types of Bots, Workloads which can be automated.

Unit- II

RPA Advanced Concepts :Standardization of processes,RPADevelopemtmethodologies,Difference from SDLC,Robotic control flow architecture,RPA business case,RPATeam,Proccess Design Document/Solution Design Document,Industries best suited for RPA,Risks& Challenges with RPA,RPA and emerging ecosystem

Installation :Installing Studio community edition,The User Interface,KeyboardShortcuts,AboutUpdating,About Automation Projects,Introduction to Automation Debugging,Managing Activation Packages,Reusing Automations Library,Installing the Chrome Extension,

Installing the Firefox Extension,Connecting your project to a source control system,Activities Guide

Variables :Managing Variables,Naming Best Practices,The Variables Panel,Generic Value Variables,TextVariables,True or False Variables,NumberVariables,ArrayVariables,Date and Time Variables,Data Table Variables,ManagingArguments,Naming Best Practices,The Arguments Panel,UsingArguments,About Imported Namespaces,Importing New Namespaces

Unit-III

Control Flow :Control Flow Introduction,If Else Statements,Loops,Advanced Control Flow,Sequences,Flowcharts,About Control Flow,Control Flow Activities,The Assign Activity,The Delay Activity,The Do While Activity,The If Activity,The Switch Activity,The While Activity,The For Each Activity,The Break Activity

Data Manipulation :Data Manipulation Introduction,Scalar variables, collections and Tables,TextManipulation,DataManipulation,Gathering and Assembling Data.

Recording and Advanced UI Interaction :Recording Introduction,Basic and Desktop Recording, WebRecording, Input/Output Methods, ScreenScraping,DataScraping,Scraping advanced techniques. Selectors: Selectors, Defining and Assessing Selectors,Customization, Debugging, Dynamic Selectors, Partial Selectors, RPA Challenge.

Unit-IV

Advanced Automation concepts and techniques: Image, Text & Advanced Citrix Automation: Introduction to Image & Text ,Automation,Image based automation,Keyboard based automation,InformationRetrieval,Advanced Citrix Automation challenges,BestPractices,Using tab for Images,Starting Apps.

Excel Data Tables & PDF :Data Tables in RPA,Excel and Data Table basics,Data Manipulation in excel,Extracting Data from PDF,Extracting a single piece of data,Anchors,Using anchors in PDF

Email Automation :Email Automation,Incoming Email automation,Sending Email automation

Exceptional Handling & Best Practices

Debugging and Exception Handling :Debugging Tools,Strategies for solving issues,Catching errors

Project Organization:What is project organization,Bestpractices,Avoidingpitfalls,Invoke Activity

Unit-V

Introduction to Orchestrator: Orchestrator: Tenants, Authentication, Users,Roles, Robots, Environments, Queues &Transactions, Schedules.

Emerging and Future Trends in IT: Emerging and Future Trends in IT: Artificial Intelligence, MachineLearning, Agentawareness,Natural Language Processing, Computer Vision

Capstone Project: Real life case studies which can be used to apply the concepts learnt during the course. The projects shall test student’s skills right from process transformation and documentation to the design and development of the actual robot.

**Course Name: Cloud AWS**

**Course Code: BCA260**

|  |  |  |  |
| --- | --- | --- | --- |
| **L (Hr.)** | **T/P (Hr.)** | **Pr (Hr.)** | **Credits** |
|  |  |  |  |

**Course Objective**

CO 1 This course helps students to build an understanding of the fundamental concepts of cloud computing.

CO2 .Students will be able to understand theoretical and practical concepts of traditional cloud and AWS cloud services.

**Syllabus**

**Unit-1**

Cloud Concepts Overview: Introduction to cloud computing, advantages of cloud, Introduction to AWS, moving to AWS Cloud.

AWS Global Infrastructure: AWS Global Infrastructure, AWS services and AWS categories. AWS management console.

**Unit-2**

Networking in AWS: Introduction, networking basics, Amazon VPC, VPC networking.

**Unit-3**

AWS Compute Services: Introduction, Computer Services Overview, Amazon EC2, Amazon Lambda and Amazon Beasnstalk.

**Unit-4**

AWS EBS: Introduction, Amazon Elastic Block Store Console, Working with EBS

**Unit-5**

AWS Cloud Architecture: Introduction, AWS Well-Architected Framework Design Principles

Automatic Scaling and Monitoring: Introduction, Amazon Elastic Load Balancing, Amazon CloudWatch, Amazon EC2 Auto Scaling.

**Course Outcomes (COs):**

**On successful completion of this course, the learner will be able to**

|  |
| --- |
| CO 1.Students are able to understand basic cloud concepts and AWS Global Infrastructure. |
| CO 2. Students are able to understand the basic concepts of networking in AWS. |
| CO 3.Students are able to understand the basic concepts of Computing Services in AWS. |
| CO 4. Students are able to understand the basic concepts of AWS EBS. |
| CO 5.Students are able to understand the basic concepts of AWS Cloud Architecture and Automatic Scaling and Monitoring. |