**ANNAMALAI UNIVERSITY**

**(Affiliated Colleges)**

**411.Master of Computer Applications**

Programme Structure and Scheme of Examination (under CBCS)

(Applicable to the candidates admitted from the academic year 2023 -2024 onwards)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Code** | **Course Title** | **Credit** | **Hours/Week** | **Examination Duration(Hrs)** | **Marks** | | | |
| **CIA** | **ESE** | | **Total** |
| **Semester-I** | | | | | | | | |
| 23PMCAC11 | Core –I Theory:  Discrete Mathematics | 4 | 6 | 3 | 25 | | 75 | 100 |
| 23PMCAC12 | Core-II Theory:  Linux and shell programming | 5 | 7 | 3 | 25 | | 75 | 100 |
| 23PMCAC13 | Core-III Theory:  Advanced Java Programming | 5 | 7 | 3 | 25 | | 75 | 100 |
| 23PMCAE14-1 | Data Engineering and Management with Lab (Practical) | 3 | 5 | 3 | 25 | | 75 | 100 |
| 23PMCAE14-2 | High Performance Computing with Lab (Practical) |
| 23PMCAE15-1 | Dot Net Technologies with lab (Practical) | 3 | 5 | 3 | 25 | | 75 | 100 |
| 23PMCAE15-2 | Architecture and Frameworks with lab (Practical) |
| **Total** | | **20** | **30** |  |  | |  | **500** |

**SEMESTER - II**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Code** | **Course Title** | **Credit** | **Hours/Week** | **Examination Duration(Hrs)** | **Marks** | | | |
| **CIA** | **ESE** | | **Total** |
| **Semester-I** | | | | | | | | |
| 23PMCAC21 | Core-IV Theory:  Data Structures and Algorithms | 5 | 6 | 3 | 25 | | 75 | 100 |
| 23PMCAC22 | Core-V Theory:  Advanced Computer Network | 5 | 6 | 3 | 25 | | 75 | 100 |
| 23PMCAP23 | Core –VI Practical: Data Structures and Algorithms Lab | 4 | 6 | 3 | 25 | | 75 | 100 |
| 23PMCAE24-1/ | Cryptography and Network Security Lab (Practical) | 3 | 4 | 3 | 25 | | 75 | 100 |
| 23PMCAE24-2 | Network Protocols Lab (Practical) |
| 23PMCAE25-1 | Computer Vision Lab (Practical) | 3 | 4 | 3 | 25 | | 75 | 100 |
| 23PMCAE25-2 | Solution Architecture Lab (Practical) |
| 23PMCAS26 | Skill Enhancement Course [SEC] – I: Software Development Technologies | 2 | 4 | 3 | 25 | | 75 | 100 |
| **Total** | | **22** | **30** |  |  | |  | **600** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | 23PMCAC11: Discrete Mathematics | **L** | **P** | **C** |
| **I** | **6** | **0** | **4** |

### 

**Course Objectives**

* To know the concepts of relations and functions
* To distinguish among different normal forms and quantifiers
* To solve recurrence relations and permutations & combinations
* To know and solve matrices , rank of matrix & characteristic equations
* To study the graphs and its types

### Unit-I

**Relations**- Binary relations-Operations on relations- properties of binary relations in a set – Equivalence relations–– Representation of a relation by a matrix -Representation of a relation by a digraph – **Functions**-Definition and examples-Classification of functions-Composition of functions-Inverse function

### Unit-II

**Mathematical Logic**-Logical connectives-**Well formed formulas** – Truth table of well formed formula –Algebra of proposition –Quine’s method- **Normal forms of well formed formulas**- Disjunctive normal form-Principal Disjunctive normal form-Conjunctive normal form-Principal conjunctive normal form-**Rules of Inference for propositional calculus** – **Quantifiers**- Universal Quantifiers- Existential Quantifiers

### Unit-III

**Recurrence Relations**- Formulation -solving recurrence Relation by Iteration- solving Recurrence Relations- Solving Linear Homogeneous Recurrence Relations of Order Two- Solving Linear Non homogeneous Recurrence Relations. **Permutations**-Cyclic permutation- Permutations with repetitions- permutations of sets with indistinguishable objects- **Combinations**- Combinations with repetition

### Unit-IV

### Matrices- special types of matrices-Determinants-Inverse of a square matrix-Cramer’s rule for solving linear equations-Elementary operations-Rank of a matrix-solving a system of linear equations-characteristic roots and characteristic vectors-Cayley-Hamilton Theorem-problems

### Unit-V

**Graphs** -Connected Graphs -Euler Graphs- Euler line-Hamiltonian circuits and paths –planar graphs – Complete graph-Bipartite graph-Hyper cube graph-Matrix representation of graphs

**Text books:**

1. N.Chandrasekaran and M.Umaparvathi, Discrete mathematics, PHI Learning Private Limited, New Delhi, 2010.

**Reference Books:**

* + - 1. Kimmo Eriksson &Hillevi Gavel, Discrete Mathematics & Discrete Models, Studentlitteratur AB, 2015.
      2. Kenneth H. Rosen Discrete Mathematics and applications, Mc Graw Hill, 2012

**Course Outcomes**

On the successful completion of the course, students will be able

| CO1: | To understand the concepts of relations and functions distinguish among normal forms | K2 | IO |
| --- | --- | --- | --- |
| CO2: | To analyze and evaluate the recurrence relations | K4,K5 | HO |
| CO3: | To distinguish among various normal forms and predicate calculus | K5 | HO |
| CO4: | To solve and know various types of matrices | K1 | LO |
| CO5: | To evaluate and solve various types of graphs | K5 | HO |

**K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAC12:Linux and Shell Programming** | **L** | **P** | **C** |
| **I** | **7** | **0** | **5** |

**Course Objectives**

* To teach principles of operating system including File handling utilities, Basic Linux commands, Scripts and filters.
* To familiarize fundamentals of shell (bash), shell programming, pipes, Control structures, arithmetic in shell interrupt processing, functions, debugging shell scripts.
* To impart fundamentals of file concepts kernel support for file, File structure related system calls (file API’s).
* To facilitate students in understanding Inter process communication, semaphore and shared memory.
* To explore real-time problem solution skills in Shell programming.

**Unit-I**

**Basic bash Shell Commands:** Interacting with the shell-Traversing the file system-Listing files and directories-Managing files and directories-Viewing file contents. **Basic Script Building:**Using multiple commands-Creating a script file-Displaying messages-Using variables-Redirecting input and output-Pipes-Performing math-Exiting the script. **Using Structured Commands:**Working with the if-then statement-Nesting ifs-Understanding the test command-Testing compound conditions-Using double brackets and parentheses-Looking at case.

(Book-1, Chapters: 3, 11, and 12)

**Unit-II**

**More Structured Commands:** Looping with for statement-Iterating with the until statement-Using the while statement-Combining loops-Redirecting loop output. **Handling User Input:** Passing parameters-Tracking parameters-Being shifty-Working with options-Standardizing options-Getting user input. **Script Control:** Handling signals-Running scripts in the background-Forbidding hang-ups -Controlling a Job-Modifying script priority-Automating script execution.

(Book-1, Chapters: 13, 14, and 16)

**Unit-III**

**Creating Functions:** Basic script functions-Returning a value-Using variables in functions-Array and variable functions-Function recursion-Creating a library-Using functions on the command line. **Writing Scripts for Graphical Desktops:** Creating text menus-Building text window widgets-Adding X Window graphics. **Introducing sed and gawk:** Learning about the sed Editor-Getting introduced to the gawk Editor-Exploring sed Editor basics.

(Book-1, Chapters: 17, 18, and 19)

**Unit-IV**

**Regular Expressions:** Defining regular expressions-Looking at the basics-Extending our patterns-Creating expressions. **Advanced sed:** Using multiline commands-Understanding the hold space-Negating a command-Changing the flow-Replacing via a pattern-Using sed in scripts-Creating sed utilities. **Advanced gawk:** Reexamining gawk-Using variables in gawk-Using structured commands-Formatting the printing-Working with functions.

(Book-1, Chapters: 20, 21, and 22)

**Unit-V**

**Working with Alternative Shells:** Understanding the dash shell-Programming in the dash shell-Introducing the zsh shell-Writing scripts for zsh.**Writing Simple Script Utilities:** Automating backups-Managing user accounts-Watching disk space. **Producing Scripts for Database, Web, and E-Mail:** Writing database shell scripts-Using the Internet from your scripts-Emailing reports from scripts. **Using Python as a Bash Scripting Alternative:** Technical requirements-Python Language-Hello World the Python way-Pythonic arguments-Supplying arguments-Counting arguments-Significant whitespace-Reading user input-Using Python to write to files-String manipulation.

(Book-1, Chapters: 23, 24, 25, and Book-2, Chapter: 14)

**Text books:**

1. Richard Blum, Christine Bresnahan, “Linux Command Line and Shell Scripting BIBLE”, Wiley Publishing, 3rd Edition, 2015.**Chapters:** 3, 11 to 14, 16 to 25.
2. Mokhtar Ebrahim, Andrew Mallett, “Mastering Linux Shell Scripting”, Packt Publishing, 2nd Edition, 2018. **Chapter:** 14.

**Reference Books:**

1. ClifFlynt,‎SarathLakshman,‎ShantanuTushar, “Linux Shell Scripting Cookbook ”, Packt Publishing, 3rd Edition, 2017.
2. Stephen G.Kochan, Patrick Wood, “Shell Programming in Unix, Linux, and OS X”, Addison Wesley Professional, 4th Edition, 2016.
3. Robert Love, “Linux System Programming”, O'Reilly Media, Inc, 2013
4. W.R. Stevens, “Advanced Programming in the UNIX environment”, 2nd Edition, Pearson Education, 2013
5. Graham Glass, King Ables, “ UNIX for Programmers and Users”, 3rd Edition, Pearson Education, 2003

**Course Outcomes**

On the successful completion of the course, students will be able

| CO1: | To understand, apply and analyze the concepts and methodology of Linux shell programming | K1-K6 |
| --- | --- | --- |
| CO2: | To comprehend, impart and apply fundamentals of control structure and script controls | K1-K6 |
| CO3: | To understand, analyses and evaluate the functions, graphical desktop interface and editors | K1-K6 |
| CO4: | To collaborate, apply and review the concepts and methodology of regular expression and advanced gawk | K1-K6 |
| CO5: | To comprehend, use and illustrate the advance concepts such as alternate shell script, data connectivity and bash scripting using python | K1-K6 |

**K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create**

**Mapping with Programme Outcomes**

|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CO1 | S | S | S | - | S | L | - | M | M | M |
| CO2 | S | S | M | - | S | L | - | M | M | M |
| CO3 | S | S | M | - | S | L | - | M | M | S |
| CO4 | S | S | M | - | S | L | - | M | M | M |
| CO5 | S | S | M | - | S | L | - | M | M | M |

**S- Strong; M-Medium; L-Low**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAC13:Advanced Java Programming** | **L** | **P** | **C** |
| **I** | **7** | **0** | **5** |

**Course Objectives**

* To gain knowledge of Object Oriented Programming Concept in Java
* To understand usages of String functions in Java
* To familiarize with the applet and swing
* To grasp the concepts on Java Beans
* To comprehend the connection between Relational Database and Java.

**Unit – I**

An Overview of Java: Object Oriented Programming- Data Types, Variables, and Arrays: Primitive Types-Literals Variables - Type Conversion and Casting- Arrays-Operators: Control Statements-Classes and Methods – Inheritance- Exception Handling.

**Unit – II**

String Handling: The String Constructors - String Length - Special String Operations - Character Extraction - String Comparison - Searching Strings - Modifying a String - Input/Output: The I/O Classes and Interfaces – File - Byte Streams - Character Streams.

**Unit – III**

The Applet Class: Basic Architecture - Applet Skeleton - Display methods - Status Window – Passing Parameters. Introducing GUI Programming with Swing– Introducing Swing - Swing Is Built on the AWT- Two Key Swing Features - The MVC Connection - Components and Containers - The Swing Packages - A Simple Swing Application - Exploring Swing.

**Unit- IV**

Java Beans: Introduction - Advantages of Beans – Introspection - The JavaBeans API - A Bean Example. Servlets: Life Cycle Simple Servlet-Servlet API-Packages-Cookies session tracking.

**Unit – V**

Network Programming: Working with URLs- Working with Sockets - Remote Method Invocation. Introduction to Database Management Systems - Tables, Rows, and Columns - Introduction to the SQL SELECT Statement - Inserting Rows - Updating and Deleting Existing Rows - Creating and Deleting Tables - Creating a New Database with JDBC - Scrollable Result Sets.

**Text Books:**

1. Herbert Schildt, “Java the Complete Reference”, 10th edition, McGraw Hill Publishing Company Ltd, New Delhi, 2017.
2. Tony Goddis, “Starting out with Java from Control Structures Through Objects” 6th Edition, Pearson Education Limited, 2016

**Reference Books:**

1. Herbert Schildt, Dale Skrien, “Java Fundamentals – A Comprehensive Introduction”, TMGH Publishing Company Ltd, New Delhi, 2013
2. John Dean, Raymond Dean, “Introduction to Programming with JAVA – A Problem Solving Approach”, TMGH Publishing Company Ltd, New Delhi,2012.

**Course Outcomes:**

On the successful completion of the course, students will be able

| CO1: | Understand the Object Oriented Program including classes and methods; inheritance and exception handling | K1-K6 |
| --- | --- | --- |
| CO2: | Complete comprehension of String functions and I/O Streams | K1-K6 |
| CO3: | Creation of graphical representation using Applet | K1-K6 |
| CO4: | Application of Servlets for designing Web based applications | K1- K6 |
| CO5: | Usage of JDBC connectivity and implementation of the concept to get desired results from database | K1-K6 |

**K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create**

**Mapping with Programme Outcomes**

|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CO1 | S | S | S | - | M | S | - | - | - | S |
| CO2 | S | S | S | - | M | S | - | - | - | L |
| CO3 | S | S | M | - | L | S | - | - | - | M |
| CO4 | M | S | M | - | S | S | - | - | - | M |
| CO5 | S | M | M | - | M | L | - | - | - | M |

**S- Strong; M-Medium; L-Low**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAE14-1:Data Engineering and Management Lab (Practical)** | **L** | **P** | **C** |
| **I** | **0** | **5** | **3** |

**Course Objectives:**

* To acquire basic scripting knowledge in MongoDB
* To learn CRUD Operation on MongoDB database
* To comprehend MongoDB using DbVisualizer
* To be familiar with Zoho CRM features
* To customize your application using Zoho CRM

**Exercises:**

1. Write a script to create a MongoDB database and perform insert operation
2. Write a MongoDB script to perform query operations
3. Write a MongoDB Script to perform update operations
4. Write a MongoDB Script to update documents with aggregation pipeline
5. Write a MongoDB script to delete single and multiple documents
6. Write a MongoDB script to perform string aggregation operations
7. Design a Data Model for MongoDB using DbVisualizer
8. Perform CRUD operations using DbVisualizer
9. Create a Zoho CRM account and organize your Tasks, Meetings and Deals
10. Create and maintain a project using Zoho CRM features

**Course Outcomes:**

On the successful completion of the course, students will be able to

| Course Outcome | Description | Knowledge Level |
| --- | --- | --- |
| **CO1** | Comprehend the scripting knowledge in MongoDB and perform basic operations in shell prompt | **K1- K6** |
| **CO2** | Implement, Create, Read, Update and Delete Operations on MongoDB database |
| **CO3** | Analyze MongoDB using DbVisualizer |
| **CO4** | Assess Zoho CRM features for managing the customer relationships |
| **CO5** | Create a customized application in Zoho CRM |

**K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6-Create**

**Mapping with Programme Outcomes**

|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CO1 | S | S | S | S | S | M | S | S | S | M |
| CO2 | S | S | S | S | S | S | S | S | S | S |
| CO3 | S | M | S | S | M | S | M | S | S | S |
| CO4 | S | S | S | M | S | S | S | L | S | S |
| CO5 | S | S | S | S | M | S | S | S | S | S |

**S- Strong; M-Medium; L-Low**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAE14-2:High Performance Computing Lab(Practical)** | **L** | **P** | **C** |
| **I** | **0** | **5** | **3** |

**Course Objectives:**

* To understand concepts of High Performance Computing.
* To get brief knowledge about PB and Slurm.
* To understand techniques of Open MP and Open MPI.
* To understand Parallel computing concepts.
* To get familiar with CUDA.

(Implemented either PB, Slurm, Open MP, OpenMPI, and CUDA)

1. Demo: - Access and best practices on HPC
2. Matrix multiplication with Job scheduling (PB or Slurm)
3. Vectors add with malloc shared
4. Vector add program with MPI
5. Hello world task for Multithreading with openMP
6. openMP shared memory on Host and Device
7. openMP Matrix Multiplication with parallelism and Barrier
8. openMP with Reduction on operands and aggregate functionality
9. Vector and Matrix multiplication on CUDA
10. Feed forward computing on CUDA

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAE15-1:Dot Net Technologies Lab (Practical)** | **L** | **P** | **C** |
| **I** | **0** | **5** | **3** |

**Course Objectives:**

* To get strong understanding of .NET Framework and C# programming.
* To get advanced programming skills in C# .NET OOPs Concepts
* To get advanced methods of manipulating data using Microsoft SQL Server.
* To get clear idea of how to developing real-time standalone, web applications using ASP .NET.
* To get clear understanding and get experience in Microsoft Azure.

**Implement the following problems using C# with Visual Studio 2017**

1. Demonstrate method overloading and method overriding
2. Class and Objects
3. Multilevel Inheritance
4. Interfaces
5. Demonstrate multiple type of Exceptions
6. Azure Storage Container Using the Microsoft Azure Storage Client Library
7. Demonstrate Read and Write a Data using Random Access Files
8. Employee management database using LINQ
9. Student management system using ASP.NET
10. Demonstrates simple Universal App.

**Course Outcomes:**

On the successful completion of the course, students will be able to,

|  |  |  |  |
| --- | --- | --- | --- |
| CO1 | Get a strong understanding of .NET Visual Studio platform | K1, K2 | LO |
| CO2 | Become a strong knowledge in C# .NET | K3 | IO |
| CO3 | Getting real-time application developing using .NET Cloud Technologies. | K4,K5 | HO |

**K1- Remember, K2 - Understand, K3 - Apply, K4 - Analyze, K5 - Evaluate, K6 -Create**

**Mapping with Programme Outcomes:**

|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CO1 | S | L | M | L | L | L | - | - | - | - |
| CO2 | S | M | L | M | M | L | - | L | - | L |
| CO3 | S | S | S | M | M | L | - | L | - | L |

* **L - Low, M- Medium, S - Strong**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAE15-2:Architecture and Frameworks – Lab (Practical)** | **L** | **P** | **C** |
| **I** | **0** | **5** | **3** |

**Course Objectives:**

* To understand and implement the basic concepts of Software architecture and its functions.
* To acquire programming skills to develop Implement various technologies and services associated with network protocols along with the challenges of data transfer.
* Implement the importance and functioning of Routing Protocols over communication service.
* To acquire skills to connect two routers and any two switches.
* To comprehend related to SSH protocols and accessing the remote device.

Note: Use the solid servers and client specification for implementation

https://github.com/solid/specification/.

**Implement the following using Linux / Windows environments**

1. Find the WebID profile document and display the necessary attributes
2. Set and access the primary authentications with account recovery mechanisms
3. Set and access the secondary authentications with account recovery mechanisms
4. Design authorization and web access control
5. Find the content representation
6. Reading resources from HTTP REST API and WebSockets API
7. Writing resources from HTTP REST API and WebSockets API
8. Data notification using Social Web App protocol
9. Managing subscriptions and friends list using Social Web App protocol
10. Managing list of followers and following list using Social Web App protocol

**Course Outcomes:**

On the successful completion of the course, students will be able to

| CO1 | Comprehend the programming skills of Software architecture tools and packages | K1-K6 |
| --- | --- | --- |
| CO2 | Understand and implement the user profiles and authentication with recovery mechanism. | K1-K6 |
| CO3 | Comprehend and evaluate the access control and content representation use of FTP server | K1-K6 |
| CO4 | Understand and implement reading and writing resources for various applications | K1-K6 |
| CO5 | Identify and examine the notifications, friends, and follower list of social application protocols. | K1-K6 |

**K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5-Evaluate, K6- Create**

**Mapping Course outcomes with Programme outcomes**

| COs | PO1 | PO2 | PO3 | PO4 | PO5 |
| --- | --- | --- | --- | --- | --- |
| CO1 | S | S | M | L | M |
| CO2 | S | M | S | S | S |
| CO3 | S | M | S | M | S |
| CO4 | S | M | L | S | M |
| CO5 | M | S | M | L | S |

**S- Strong; M-Medium; L-Low**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAC21: Data Structures and Algorithms** | **L** | **P** | **C** |
| **II** | **6** | **0** | **5** |

**Course Objectives:**

* To get a clear understanding of various ADT structures.
* To understand how to implement different ADT structures with real-time scenarios.
* To analyze the various data structures with their different implementations.
* To get an idea of applying right models based on the problem domain.
* To realize, and understand how and where to implement modern data structures with Python language.

**Unit-I**

**Abstract Data Types:** Introduction-Date Abstract Data Type-Bags-Iterators. **Arrays**: Array Structure-Python List-Two Dimensional Arrays-Matrix Abstract Data Type. **Sets, Maps:** Sets-Maps- Multi-Dimensional Arrays.

**Unit-II**

**Algorithm Analysis:** Experimental Studies-Seven Functions-Asymptotic Analysis. **Recursion:** Illustrative Examples-Analyzing Recursive Algorithms-Linear Recursion- Binary Recursion-Multiple Recursion.

**Unit-III**

**Stacks, Queues, and Deques:** Stacks- Queues- Double-Ended Queues Linked. **Lists:** Singly Linked Lists-Circularly Linked Lists-Doubly Linked Lists. **Trees:** General Trees-Binary Trees-Implementing Trees-Tree Traversal Algorithms.

**Unit-IV**

**Priority Queues:** Priority Queue Abstract Data Type- Implementing a Priority Queue- Heaps-Sorting with a Priority Queue. **Maps, Hash Tables, and Skip Lists:** Maps and Dictionaries-Hash Tables- Sorted Maps-Skip Lists-Sets, Multisets, and Multimaps.

**Unit-V**

**Search Trees:** Binary Search Trees-Balanced Search Trees-AVL Trees-Splay Trees. **Sorting and Selection:** Merge sort-Quick sort-Sorting through an Algorithmic Lens- Comparing Sorting Algorithms-Selection. **Graph Algorithms:** Graphs-Data Structures for Graphs-Graph Traversals-Shortest Paths-Minimum Spanning Trees.

**Text books:**

1. Rance D. Necaise, “Data Structures and Algorithms Using Python”, John Wiley & Sons, 2011. (Unit – 1)**Chapters:** 1, 2, 3.
2. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, “Data Structures and Algorithms in Python”, John Wiley & Sons, 2013. (Unit – 2, 3, 4, and 5)**Chapters:** 3 to 12, and 14.

**Reference Books:**

1. Dr. Basant Agarwal; Benjamin Baka, “Hands-On Data Structures and Algorithms with Python: Write complex and powerful code using the latest features of Python 3.7”, Packt Publishing, 2018.
2. Magnus Lie Hetland, “Python Algorithms: Mastering Basic Algorithms in the Python Language”, Apress, 2014.

**Course Outcomes:**

On the successful completion of the course, students will be able to,

| CO1 | Understand various ADT concepts | K1-K6 |
| --- | --- | --- |
| CO2 | Familiar with implementation of ADT models with Python language and understand how to develop ADT for the various real-time problems |
| CO3 | Apply with proper ADT models with problem understanding |
| CO4 | Apply and Analyze right models based on the problem domain |
| CO5 | Evaluate modern data structures with Python language |

K1- Remember, K2 - Understand, K3 - Apply , K4 - Analyze, K5 - Evaluate, K6 -Create

**Mapping with Programme Outcomes:**

|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CO1 | S | M | L | L | L | L | S | S | S | L |
| CO2 | S | M | S | M | M | L | L | L | L | L |
| CO3 | S | S | S | L | L | L | M | M | M | M |
| CO4 | S | S | S | L | L | L | M | M | M | L |
| CO5 | S | S | S | L | M | M | S | S | S | S |

**L - Low, M- Medium, S - Strong**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAC22: Advanced Computer Network** | **L** | **P** | **C** |
| **II** | **6** | **0** | **5** |

**Course Objectives:**

* Study the advanced concepts of the computer networking and enumerate the layers and TCP/IP model.
* Acquire knowledge of Wireless communication and Data link layer.
* Understand the datalinks and protocols.
* Gain core knowledge of Network layer routing protocols and IP addressing.
* Study the transport layer and network security

**Unit 1 - Introduction –** Network Hardware – Software – Reference Models – OSI and TCP/IP models – Example networks: Internet, 3G Mobile phone networks, Wireless LANs –RFID and sensor networks - Physical layer – Theoretical basis for data communication - guided transmission media

**Unit-2-Wireless transmission** – Communication Satellites–Digital modulation and multiplexing - Telephones network structure – local loop, trunks and multiplexing, switching. Data link layer: Design issues–error detection and correction.

**Unit 3-Elementary data link protocols**- sliding window protocols – Example Data Link protocols–Packet over SONET, ADSL-Medium Access Layer–Channel Allocation Problem–Multiple Access Protocols.

**Unit 4-Network layer**- design issues - Routing algorithms - Congestion control algorithms –Quality of Service – Network layer of Internet- IP protocol – IP Address – Internet Control Protocol.

**Unit 5-Transport layer** – transport service- Elements of transport protocol - Addressing, Establishing & Releasing a connection – Error control, flow control, multiplexing and crash recovery- Internet Transport Protocol–TCP- Network Security: Cryptography.

## Text Books:

1.Andrew S Tanenbaum(2018),“ComputerNetworks”,LowpriceEdition,FourthEdition.

## Reference Books:

1. Teresa C. Piliouras (2015), “Network Design Management and Technical Perspectives”, Auerbach Publishers, Second Edition.
2. https:// [www.pdfdrive.com/ advanced-computing-networking- and-informatics- volume-1-](http://www.pdfdrive.com/%20advanced-computing-networking-%20and-informatics-%20volume-1-)advanced- computing- and- informatics- proceedings- of-the-second-international-conference-on-advanced-computing - networking- and- informatics- icacni- 2014-e174550006 .html

**Course Outcomes (COs):**

At the end of the course, the student will be able to

|  |  |
| --- | --- |
| **CO1** | Understand the terminology and concepts of the OSI reference model |
| **CO2** | Student will get the knowledge of protocols, network interfaces, and design issues in local area networks and wide area networks. |
| **CO3** | Understand wireless networking concepts, and be familiar with contemporary issues in networking technologies. |
| **CO4** | Gain knowledge the network tools and network programming. |
| **CO5** | Understand the Establishing and Releasing a connection in transport layer. |

## Outcome Mapping:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CO/PO** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** |
| **CO1** |  |  | S |  | M | S |  |  | M | M |
| **CO2** |  |  |  | L | S | M |  | S |  | M |
| **CO3** | S |  |  | S |  | S |  | M |  | L |
| **CO4** | S |  | S | M | M | S |  | L |  | S |
| **CO5** |  | S |  |  |  |  | S |  | M |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAP23 : Data Structures and Algorithms Lab** | **L** | **P** | **C** |
| **II** | **0** | **6** | **4** |

**Course Objectives:**

* To understand Stack, Queue and Doubly Linked ADT structures.
* To implement different ADT structures with real-time scenarios.
* To analyze the recursion concepts.
* To apply different sorting and tree techniques.
* To implement modern data structures with Python language.

**Implement the following problems using Python 3.4 and above**

1. Recursion concepts.
2. Linear recursion
3. Binary recursion.
4. Stack ADT.
5. Queue ADT.
6. Doubly Linked List ADT.
7. Heaps using Priority Queues.
8. Merge sort.
9. Quick sort.
10. Binary Search Tree.
11. Minimum Spanning Tree.
12. Depth First Search Tree traversal.

**Course Outcomes:**

On the successful completion of the course, students will be able to,

| CO1 | Strong understanding in various ADT concepts | K1-K6 |
| --- | --- | --- |
| CO2 | To become a familiar with implementation of ADT models |
| CO3 | Apply sort and tree search algorithms |
| CO4 | Evaluate the different data structure models |
| CO5 | Learn how to develop ADT for the various real-time problems |

K1- Remember, K2 - Understand, K3 - Apply , K4 - Analyze, K5 - Evaluate, K6 -Create

**Mapping with Programme Outcomes:**

|  | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CO1** | S | M | L | L | L | L | S | S | M | M |
| **CO2** | S | M | S | M | M | L | S | M | S | L |
| **CO3** | S | S | S | L | L | L | M | M | M | M |
| **CO4** | S | S | S | M | M | S | M | M | S | S |
| **CO5** | S | S | S | S | L | M | S | M | M | M |

**L - Low, M- Medium, S – Strong**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAE24-1: Cryptography and Network Security Lab**  **(Practical)** | **L** | **P** | **C** |
| **II** | **0** | **4** | **3** |

**Course Objective**

* To develop in classical encryption techniques and advanced encryption standards.
* To acquire programming skills in Implement various cryptographic algorithms including secret key cryptography.
* To develop hashes, message digests and public key algorithms.
* Implement different encryption and decryption techniques.
* To comprehend related to confidentiality and authentication techniques.

**Implement the following**

1. Write a program that contains a string (char pointer) with a value ‘Hello world’. The program should XOR each character in the string with 0 and display the result
2. Write a program to perform encryption and decryption using the Ceaser Cipher
3. Write a program to perform encryption and decryption using the Hill Cipher
4. Write a program to perform encryption and decryption using the Substitution Cipher
5. Write a program to perform encryption and decryption using the DES algorithm
6. Connect to switch with a computer and enable the port security
7. Defeating malware using Building Trojans and Rootkit hunter
8. Implement signature scheme – Digital Signature Standard
9. Identify and capture the user name and password in a same network using wires hark
10. Implement Man-in-the-middle attack and Session hijacking

**Course Outcomes:**

On the successful completion of the course, students will be able to

| CO1 | Comprehend the programming skills in classical encryption techniques and to develop advanced encryption standards | K1-K6 |
| --- | --- | --- |
| CO2 | Understand and implement the various cryptographic algorithms including secret key cryptography, hashes and message digests | K1-K6 |
| CO3 | Evaluate the use of different encryption and decryption techniques | K1-K6 |
| CO4 | Design to Solve related confidentiality and authentication problems | K1-K6 |
| CO5 | Create public key algorithms | K1-K6 |

**K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5-Evaluate, K6- Create**

**Mapping Course outcomes with Programme outcomes**

| COs | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CO1 | S | - | - | - | - | L | - | - | - | - |
| CO2 | S | - | M | - | M | L | - | - | - | - |
| CO3 | S | - | S | - | S | L | - | - | - | S |
| CO4 | S | - | S | - | S | L | - | - | - | S |
| CO5 | S | - | S | - | S | L | - | - | - | S |

**S- Strong; M-Medium; L-Low**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAE24-2 : Network Protocols Lab**  **(Practical)** | **L** | **P** | **C** |
| **II** | **0** | **4** | **3** |

**Course Objectives:**

* To understand and implement the basic concepts of Transmission Control Protocol/Internet Protocol and associated functions.
* To acquire programming skills in Implement various technologies and services associated with network protocols along with the challenges of data transfer.
* Implement the importance and functioning of Routing Protocols over communication service.
* To acquire skills to connect two routers and any two switches.
* To comprehend related to SSH protocols and accessing the remote device.

**Implement the following using Linux / Windows environments**

1. Implement the following commands
   1. ipconfig
   2. ping
   3. traceroute
   4. netsat
   5. nslookup
2. Implement the following server commands
   1. ifconfig
   2. ip
   3. tracepath
   4. ss
   5. tcpdum
3. Connect and place the given file in the FTP server
4. Install packet tracer and connect a computer to router, switch and get a Icmp request
5. Implement the SSH protocols and accessing the remote device
6. Connect any two switches and get the status of each switches
7. Connect two routers and get packets from the routers.
8. Get the access of the router by connecting with working computer
9. Identify the route password of server and get the connection using telnet
10. Install wire shark for capture and analyse the packets (TCP /UDP)

**Course Outcomes:**

On the successful completion of the course, students will be able to

| CO1: | Comprehend the programming skills the SSH protocols and accessing the remote device | K1-K6 |
| --- | --- | --- |
| CO2: | Understand and implement the various functioning of Routing Protocols over communication service. | K1-K6 |
| CO3: | Evaluate the use of FTP server | K1-K6 |
| CO4: | Design to Connect any two switches and get the status of each switches | K1-K6 |
| CO5: | Solve to Connect two routers and get packets from the routers. | K1-K6 |

**K1- Remember, K2- Understand, K3- Apply , K4- Analyze, K5-Evaluate, K6- Create**

**Mapping Course outcomes with Programme Outcomes**

| COs | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CO1 | S | - | - | - | - | L | - | - | - | - |
| CO2 | S | - | M | - | M | L | - | - | - | - |
| CO3 | S | - | S | - | S | L | - | - | - | S |
| CO4 | S | - | S | - | S | L | - | - | - | S |
| CO5 | S | - | S | - | S | L | - | - | - | S |

**S- Strong; M-Medium; L-Low**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAE25-1: Computer Vision Lab**  **(Practical)** | **L** | **P** | **C** |
| **II** | **0** | **4** | **3** |

**Course Objectives:**

* To get an idea of how to build a computer vision application with Python language.
* To learn the basic image handling and processing
* To get familiar with various Computer Vision fundamental algorithms and how to implement and apply.
* To get an idea of how to implement the image transforms.
* To understand various image segmentation algorithms.

**Implement the following problems using Python with OpenCV**

1. Image Loading, Exploring, and displaying an Image.
2. Access and Manipulate of Image Pixels.
3. Image Transformations.
4. Resizing
5. Rotation
6. Addition operation of Two Images.
7. Image filtering operations
8. Mean Filtering
9. Gaussian Filtering
10. Image Binarization Using Simple Thresholding method.
11. Edge Detection operation using Sobel and Scharr Gradients.
12. Find Grayscale and RGB Histograms of an Image.
13. Segment an Image using K-means Clustering algorithm.
14. Write a program to classify an Image using KNN Classification algorithm.

**Course Outcomes:**

On the successful completion of the course, students will be able to,

|  |  |  |
| --- | --- | --- |
| CO1 | To develop and implement the image loading and exploring | K1-K6 |
| CO2 | To Evaluate the image transforms |
| CO3 | To apply and analyze for image processing denoising algorithms |
| CO4 | To design and develop the Image Segmentation using Edge detection and Histograms |
| CO5 | To apply and analyze image clustering and classification algorithms |

**K1- Remember, K2 - Understand, K3 - Apply , K4 - Analyze, K5 - Evaluate, K6 -Create**

**Mapping with Programme Outcomes:**

|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CO1 | S | L | M | L | L | L | M | M | M | S |
| CO2 | S | M | L | M | M | L | S | L | S | L |
| CO3 | S | S | S | M | M | L | M | L | M | L |
| CO4 | S | S | S | M | M | L | M | L | M | L |
| CO5 | S | S | S | M | M | L | S | L | S | L |

**L - Low, M- Medium, S - Strong**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAE25-2: Solution Architecture Lab**  **(Practical)** | **L** | **P** | **C** |
| **II** | **0** | **4** | **3** |

**Course Objectives:**

* To get familiar with AWS cloud services
* To design secure, scalable, and well structured cloud solutions
* To create EC2 instance and configure options
* To connect EC2 with Linux instance
* To connect VPN server to securely access instances

**Implement the following problems:**

1. Managing Virtual Private Cloud
2. Creating and Configuring Internet Gateways
3. Configuring Routing Tables
4. Working with Amazon Elastic Cloud Compute (EC2)
5. Connecting EC2 Linux instance using PuTTY,Gitbash and Console
6. Recovering and connecting EC2 instance if the SSH key is lost
7. Creating and Configuring Elastic Load Balancer
8. Scheduling Auto Snapshot of volumes
9. Configuring Centralized Log Management using CloudWatch Log
10. Connecting OpenVPN server

**Course Outcomes:**

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

|  |  |  |
| --- | --- | --- |
| **CO1:** | Understand AWS cloud services and manage the cloud data | **K1- K6** |
| **CO2:** | Develop secure, scalable, and well structured cloud solutions |
| **CO3:** | Implement EC2 instance and configure the instance |
| **CO4:** | Connect EC2 with Linux instance and perform operations |
| **CO5:** | To connect VPN server to access instances with more security |

**K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create**

**Mapping with Programme Outcomes**

|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CO1 | S | S | S | S | S | S | S | S | S | M |
| CO2 | S | S | S | S | S | S | S | S | S | S |
| CO3 | S | M | S | S | S | S | M | S | S | S |
| CO4 | S | S | S | M | S | S | S | S | S | S |
| CO5 | S | S | S | S | M | S | S | S | S | M |

**S- Strong; M-Medium; L-Low**

**Skill Enhancement Course [SEC]-I**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semester** | **23PMCAS26: Software Development Technologies** | **L** | **P** | **C** |
| **II** | **4** | **0** | **2** |

**Course Objectives:**

* To learn and Implementing Micro services
* To analysing the Azure Kubernetes Service
* To learn and anlyse .NET DevOps for Azure and its applications
* To building code for .NET core applications
* To get familiarized with Azure pipelines

**Unit -I**

**Implementing Microservices :** Client to microservices communication, Interservice communication, data considerations, security, monitoring, microservices hosting platform options. Azure Service Fabric: Introduction, core concepts, supported programming models, service fabric clusters, develop and deploy applications of service fabric. Monitoring Azure Service Fabric Clusters: Azure application, resource manager template, Adding Application Monitoring to a Stateless Service Using Application Insights, Cluster monitoring, Infrastructure monitoring.

**Unit-II**

**Azure Kubernetes Service (AKS) :** Introduction to kubernetes and AKS, AKS development tools, Deploy applications on AKS. Monitoring AKS: Monitoring, Azure monitor and analytics, monitoring AKS clusters, native kubernetes dashboard, Prometheus and Grafana. Securing Microservices: Authentication in microservices, Implenting security using API gateway pattern, Creating application using Ocrlot and securing APIs with Azure AD. Database Design for Microservices: Data stores, monolithic approach, Microservices approach, harnessing cloud computing, database options on MS Azure, overcoming application development challenges. Building Microservices on Azure Stack: Azure stack, Offering IaaS, PaaS on-premises simplified, SaaS on Azure stack.

**Unit-III**

**.NET DevOps for Azure:** DevOps introduction, Problem and solution. Professional Grade DevOps Environment: The state of DevOps, professional grade DevOps vision, DevOps architecture, tools for professional DevOps environment, DevOps centered application. Tracking work: Process template, Types of work items, Customizing the process, Working with the process. Tracking code: Number of repositories, Git repository, structure, branching pattern, Azure repos configuration, Git and Azure.

**Unit-IV**

**Building the code:** Structure of build, using builds with .NET core and Azure pipelines, Validating the code: Strategy for defect detection, Implementing defect detection. Release candidate creation: Designing release candidate architecture, Azure artifacts workflow for release candidates, Deploying the release: Designing deployment pipeline, Implementing deployment in Azure pipelines. Operating and monitoring release: Principles, Architectures for observability, Jumpstarting observability.

**Unit-V**

**Introduction to APIs:** Introduction, API economy, APIs in public sector. API Strategy and Architecture: API Strategy, API value chain, API architecture, API management. API Development: Considerations, Standards, kick-start API development, team orientation. API Gateways: API Gateways in public cloud, Azure API management, AWS API gateway. API Security: Request-based security, Authentication and authorization.

**Text Books:**

1. Harsh Chawla and Hemant Kathuria, Building Microservices Applications on Microsoft Azure- Designing, Developing, Deploying, and Monitoring, Apress, 2019.
2. Jeffrey Palermo , NET DevOps for Azure A Developer’s Guide to DevOps Architecture the Right Way, Apress, 2019.
3. Thurupathan and Vijayakumar, Practical API Architecture and Development with Azure and AWS - Design and Implementation of APIs for the Cloud, Apress, 2018.

**Reference Books:**

1. Karl Matthias and Sean P. Kane, Docker: Up and Running, O'Reilly Publication, Second Edition 2018.
2. Len Bass,IngoWeber, LimingZhu,”DevOps, A Software Architects Perspective”, AddisonWesley-Pearson Publication, First Ediiton 2015.
3. John Ferguson Smart,”Jenkins, The Definitive Guide”, O'Reilly Publication, First Ediiton 2011.

**Course Outcomes:**

On the successful completion of the course, students will be able to

| CO1: | To understand, apply and summarize the basic concepts of  Micro services communication Microsoft Azure and Dev Ops for software development life cycle | K1-K6 |
| --- | --- | --- |
| CO2: | To illustrate, and implement Azure Kubernetes Service tools for software development life cycle | K1-K6 |
| CO3: | To recognize, analyse and summarize the functionalities of .NET Dev Ops for Azure applications | K1-K6 |
| CO4: | To understand, design and evaluate the principles and architecture service tools for software development life cycle. | K1-K6 |
| CO5: | To comprehend, implement and review the functionalities of API and API gateways for cloud and Azure applications | K1-K6 |

**K1- Remember, K2- Understand, K3- Apply, K4- Analyze, K5- Evaluate, K6- Create**

**Mapping with Programme Outcomes**

|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CO1 | S | S | M | - | M | - | - | - | S | - |
| CO2 | S | S | M | - | M | - | - | - | S | - |
| CO3 | S | S | S | - | S | - | - | - | S | S |
| CO4 | S | S | M | - | M | - | - | - | S | - |
| CO5 | S | S | M | - | M | - | - | - | S | - |

**S- Strong; M-Medium; L-Low**