SANJEEV AGRAWAL GLOBAL EDUCATIONAL UNIVERSITY, BHOPAL

Syllabus

for

Bachelor of Computer Application (BCA)

Discipline Core Courses

Batch 2023
(Under NEP 2020)

School of Computer Application

wef 2023-24

COURSE CODE	MATHEMATICS	Total Lecture: 60 Theory:45 Tutorial:15
CA23DC001		(LTP=3-1-0=4)

Course Objectives

- To understand the basic concepts of algebra, Linear Equations and Linear Dependence.
- To apply the De'Moivre's theorem and its application to find roots of algebraic equations
- To develop skills in Differential Calculus and Expansion of functions.
- To understand the concept of partial differentiation and its application

UNIT	CONTENTS	HOURS
I	**Matrix Algebra and its applications** Rank of Matrix, Normal form of matrix, PAQ form of matrix, System of Linear Equations, Linear Dependence and Independence, Linear Transformations. Eigen values, Eigen Vectors, Cayley – Hamilton Theorem.	7
II	**Complex Algebra and its applications** Argand's Diagram, De'Moivre's theorem and its application to findroots of algebraic equations. Hyperbolic Functions, Inverse Hyperbolic Functions, Logarithm ofComplex Numbers.	9
III	**Differential Calculus& Expansion of functions** Taylor's Series and Maclaurin's Series, IndESErminate Forms (all various forms), L' Hospital's Rule, Evaluation of Limits	9
IV	**Partial differentiation** Partial Derivatives, Euler's Theorem on homogeneous functions, Total Derivatives & Implicit functions, Change of independent variables.	10
V	**Application of partial differentiation** Errors and Approximations, Maxima & Minima for two and several variables, Jacobians and their applications.	10

Course Outcomes as per Bloom's Taxonomy

At the end of the course the students should be able to:

CO1	**Understand** the basic concepts of algebra, Linear Equations and linear dependence.	
CO2	**Apply** De'Moivre's theorem to solve algebraic equations and geometrical expressions of the complex function.	
CO3	**Solve** the functions using Taylor's Series and Maclaurin's Series expansion	
CO4	**Apply** the concepts of partial differentiation like maxima and minima for two and several variables, Jacobians and their applications.	
CO5	**Analyze** the applications of partial differentiation	

Text Book	• Grewal, B. S., & Grewal, J. S. (1996**). Higher engineering mathematics**. *2002, Khanna Publishers, New Delhi*. • Ramana, B. V. (2006). **Higher Engineering Mathematics**. Tata McGraw-Hill Education.
Reference Books	• Dass, H. K. (2008). **Advanced engineering mathematics**. S. Chand Publishing. • Rutland, L. W. (1963). **Advanced Engineering Mathematics**. Erwin Kreyszig. Wiley, New York. • O'neil, P. V. (2017). **Advanced engineering mathematics**. Cengage learning. • Greenberg, M. D. (1998). **Advanced Engineering Mathematics**. Pearson Education India.

COURSE CODE	Data Structures and Algorithms	Total: 45 hrs Lecture: 15 Practical: 30
CA23DC002		2– 0– 4

Course Objectives:

- Demonstrate the ability to design, implement, and evaluate algorithms to solve a wide range of computational problems.
- Analyze the time and space complexity of different algorithms and make informed decisions about their suitability for specific tasks.
- Implement and manipulate linear data structures efficiently using appropriate algorithms and techniques.
- Apply tree traversal algorithms, such as breadth-first and depth-first traversal, to explore and analyze tree structures.
- Implement graph data structures and operations, including vertex and edge manipulation.

UNIT	CONTENT	HOURS
1	**Fundamentals of Algorithms** Algorithm definitions, Asymptotic notations, O-notation, Omega notation and theta notation. Time complexity and space complexity, Average and worst-case analysis.	6
2	**Types of algorithms** Sorting and searching algorithms, Divide and conquer algorithms, Greedy algorithms, Dynamic programming, Graph Algorithms, String matching, The class P and NP problems.	6
3	**Linear Data Structures:** Arrays and its operations. Stacks: LIFO structure, create, POP, PUSH, delete stack. Queues: FIFO structure Priority Queues, Circular Queues, operations on Queues. Linked Lists: Nodes, Linked List operations: Create List, Insert Node (empty list, beginning, Middle, end), Delete node (First, general case), Search list, Retrieve Node, add node, remove node, Print List.	6
4	**Trees:** Introduction to Trees, Binary Trees :Travesals (breadth-first, depth-first), Expression Trees: Infix, Prefix, Postfix Traversals. Search Trees, Binary Search Trees, B Trees, AVL trees. Heaps: Structure, Basic algorithms – Reheap Up, Reheap Down, build heap, Insert, Delete.	6
5	**Graphs** Terminology, Operations: Add vertex, delete vertex, Add Edge, Delete Edge, Find vertex, Traverse Graph: Depth-First, Breadth-First. Graph Storage Structures: Adjacency Matrix, Adjacency List.	6

Course Outcomes (as per Bloom's Taxonomy)

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

CO 1	Compare the efficiency and performance of algorithms based on worst-case, average-case, and best-case scenarios, enabling the selection of the most suitable algorithm for a given problem.
CO 2	Demonstrate various algorithmic techniques, such as sorting, searching, dividing and conquer, greedy, and dynamic programming, to solve computational problems efficiently and effectively.
CO 3	Implement algorithms that leverage linear data structures, such as arrays, stacks, queues, and linked lists, to solve real-world problem in areas like data storage, text processing, and resource management.
CO 4	Develop efficient algorithms for tree operations, such as insertion, deletion, and searching, enabling effective manipulation and utilization of tree-based data structures in various applications.
CO 5	Solve graph-related problems, such as finding shortest paths, determining network connectivity, and constructing minimum spanning trees, by utilizing appropriate graph traversal algorithms and data structures.
Text Books	• "Data Structures and Algorithms in C" by Michael T. Goodrich, Roberto Tamassia, and David M. Mount (Publication Date: 2018). • "Data Structures and Algorithms Made Easy: Data Structures and Algorithmic Puzzles" by Narasimha Karumanchi (Publication Date: 2017).
Reference Books	• "Data Structures and Algorithms: A First Course" by T. R. Padmanabhan (Publication Date: 2018). • "C Programming for the Absolute Beginner, Second Edition" by Michael Vine (Publication Date: 2018).

List of Experiments:

1. Write a program in C to calculate the time and space complexity of a given algorithm. The program should take input as the size of the input data and measure the execution time and space used by the algorithm for different input sizes.
2. Write a program in the programming language in C to compare the performance of different sorting algorithms. The program should implement at least three sorting algorithms (e.g., bubble sort, insertion sort) and measure their

execution time for different input sizes.

3. Write a program in C to implement a divide-and-conquer algorithm for finding the maximum element in an array. Your program should include the following-
   - Implement a recursive function that takes an array of integers, starting index, and ending index as input and returns the maximum element in the array.
   - Divide the array into smaller subarrays and recursively find the maximum element in each subarray.
   - Combine the results from the subarrays to find the maximum element in the original array.
   - Test your program with different arrays and analyze its performance.
   - Display the input array and the maximum element found.

4. Write a program in C to implement and perform various operations on a stack. The program should allow the user to create a stack, push elements onto the stack, pop elements from the stack, and display the contents of the stack.

5. Write a program in C to implement and perform various operations on a singly linked list. The program should allow the user to create a linked list, insert nodes at different positions, delete nodes, search for a specific value, retrieve nodes, add nodes, remove nodes, and print the contents of the linked list.

6.Write a program in C to implement the operations of a Binary Search Tree (BST). Your program should include the following functionalities:
   - Create a structure to represent a node in a BST, containing an integer key and left and right child pointers.
   - Implement a function to create a new node and initialize its key and child pointers.
   - Implement functions to perform the following operations:
   - Insert a node with a given key into the BST.
   - Delete a node with a given key from the BST.
   - Search for a node with a given key in the BST.
   - Traverse the BST using both breadth-first and depth-first traversal techniques.
   - Display the BST in different traversal orders, such as in-order, pre-order, and post-order.

7. Write a program in C to implement various operations on a graph. Your program should include the following functionalities:
   - Create a structure to represent a graph, containing variables for the number of vertices and edges, and data structures to store the graph using either an adjacency matrix or an adjacency list.
   - Implement functions to perform the following operations:
   - Add a vertex to the graph.
   - Delete a vertex from the graph.
   - Add an edge between two vertices.
   - Delete an edge between two vertices.
   - Find a vertex in the graph.
   - Traverse the graph using both depth-first and breadth-first traversal techniques.

8. Perform the following on arrays-
   a) Count the frequency of each element of an array.
   b) Count the total number of duplicate elements in an array.
   c) Insert a new value in the array (sorted list).

9. Write C programs for the following-
   a. Finding Factorial – Iterative Approach and Recursive Approach
   b. Printing Fibonacci Series – Iterative Approach and recursive approach

10. Write a C Program to Reverse a Stack using Recursion.

11. Write a C Program to Implement Two Stacks using a Single Array & Check for Overflow & Underflow.

12. Write a C Program to Check String is Palindrome using Stack.

13. Write a C Program to Identify whether the String is Palindrome or not using Stack.

14. Write a C Program to Implement Queues using Stacks.

15. Write a program to implement AVL tree.

SANJEEV AGRAWAL GLOBAL EDUCATIONAL UNIVERSITY, BHOPAL

Syllabus

for

Bachelor of Computer Application (BCA)

Cloud Computing (Major Specialization)

Batch 2023
(Under NEP 2020)

School of Computer Application

wef 2023-24

COURSECODE	INTRODUCTION TO CLOUD COMPUTING AND DATACENTER	Total: 45 hrs Lecture: 15 Practical: 30
CA23CC001		Course Credit: 3-0-2

Course Objectives:

- Understand the fundamentals of cloud computing, including its history, characteristics, and the need for its adoption.
- Explore the advantages and possible disadvantages of cloud computing, and evaluate its potential impact on business processes.
- Examine the various cloud deployment models, such as public, private, hybrid, and community clouds, and understand their differences and use cases.
- Gain knowledge of different cloud delivery models, including Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS), and analyze their benefits, risks, and examples.
- Familiarize yourself with popular cloud platforms, such as Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform, and Salesforce.com, and understand their features, evaluation criteria, and impact.

UNIT	CONTENT	HOURS
1	**Fundamentals of Cloud Computing:** Cloud Computing Basics – History of Cloud Computing, Characteristics of Cloud Computing, Need for Cloud computing, Advantages and Possible Disadvantages of cloud computing, Cloud Deployment Models – Public, Private, Hybrid, Community, Other deployment Models. Evolving Data Center into Private Cloud, Datacenter Components, Extracting Business value in Cloud Computing – Cloud Security, Cloud Scalability, Time to Market, Distribution over the Internet, Cloud Computing Case Studie.	9
2	**Cloud Delivery Models** Introduction to Cloud Services, **Infrastructure as a Service (IaaS)** – Overview, Virtualization, Container, Pricing Models, Service Level Agreements, Migrating to the Cloud, IaaS Networking options, Virtual Private Cloud (VPC), IaaS Storage – File and Object storage, Data Protection, IaaS security, Benefits, Risks and Examples of IaaS. **Platform as a Service (PaaS)** – Overview, IaaS vs PaaS, PaaS Examples, benefits and risks. **Software as a Service (SaaS)** – Introducing SaaS, SaaS Examples – Office 365, Google G Suite, Salesforce.com, Evaluating SaaS – user and vendor perspective, Impact of SaaS, Benefits and risks of SaaS. Other Services on Cloud, Cloud Delivery Models Considerations.	9
3	Introducing Cloud Platforms, Evaluating cloud platforms, Cloud Platform technologies – Amazon Web Services, Microsoft Azure, Google Cloud Platform, Salesforce.com, Impact of Cloud platforms. Private Cloud Platforms – Introducing Private clouds – Microsoft Azure stack, Open stack, AWS Greengrass, Impact of Private clouds **Cloud Migration**: Delivering Business Processes from the Cloud: Business process examples, Broad Approaches to Migrating into the Cloud, The Seven-Step Model of Migration into a Cloud, Efficient Steps for migrating to cloud., Risks: Measuring and assessment of risks, Company concerns Risk Mitigation methodology for Cloud computing, Case Studies	9
4	**Overview of Data Centers** Data Centers Defined, Data Center Goals, Data Center Facilities, Roles of Data Centers in the Enterprise, Roles of Data Centers in the Service Provider Environment, Application Architecture Models. The Client/Server Model and Its Evolution, The n-Tier Model, Multitier Architecture Application Environment, Data Center Architecture.	9
5	**Data Center Requirements** Data Center Prerequisites, Required Physical Area for Equipment and Unoccupied Space, Required Power to Run All the Devices, Required Cooling and HVAC, Required Weight, Required Network Bandwidth, Budget Constraints, selecting a Geographic Location, Safe from Natural Hazards, Safe from Man-Made Disasters, Availability of Local Technical Talent, Abundant and Inexpensive Utilities Such as Power and Water, Selecting an Existing Building (Retrofitting), tier standard.	9

Test Books	• "Cloud Computing: Concepts, Technology & Architecture" by Thomas Erl, Ricardo Puttini, and Zaigham Mahmood. • "Cloud Computing: A Practical Approach" by Anthony T. Velte, Toby J. Velte, and Robert Elsenpeter • "Cloud Computing: Principles and Paradigms" by Rajkumar Buyya, James Broberg, and Andrzej Goscinski.
Reference Books	• "Cloud Computing: From Beginning to End" by Ray J. Rafaels. • "The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines" by Luiz André Barroso and Urs Hölzle.

List of Experiments:

1. Exploring Cloud Computing Services: Sign up for a free trial account with a cloud service provider such as Amazon Web Services (AWS), Microsoft Azure, or Google Cloud Platform. Familiarize yourself with the service offerings and navigate through the console.
2. Creating a Virtual Machine (VM) on IaaS: Provision a virtual machine on an IaaS platform like AWS EC2 or Azure VM. Configure the instance specifications, select an operating system, and launch the VM.
3. Deploying a Web Application on PaaS: Use a PaaS platform like Heroku or Google App Engine to deploy a simple web application. Follow the platform's documentation to create the necessary files, set up the application, and deploy it to the cloud.
4. Exploring SaaS Applications: Sign up for a trial account with a popular SaaS application like Office 365 or Salesforce.com. Explore the features and functionality offered by the application from both a user and vendor perspective.
5. Migrating a Local Server to IaaS: Select an existing application or server from your local environment and migrate it to an IaaS platform. This could involve creating a VM, configuring the server, transferring data, and ensuring the application runs successfully in the cloud.
6. Implementing Cloud Storage: Set up and configure cloud storage services such as AWS S3 or Azure Blob Storage. Upload files, create buckets/containers, and explore the storage options and features available.
7. Building a Private Cloud Environment: Install and configure an open-source private cloud platform like OpenStack on a local machine or a virtual environment. Explore the features and capabilities of the private cloud platform.
8. Evaluating Cloud Platforms: Compare and evaluate different cloud platforms (e.g., AWS, Azure, Google Cloud Platform) based on criteria such as pricing, scalability, available services, and management tools. Create a report highlighting the strengths and weaknesses of each platform.
9. Assessing Cloud Security: Research and identify best practices for cloud security. Perform a security assessment on a cloud environment, identify potential vulnerabilities, and propose mitigation strategies.
10. Cloud Migration Planning: Select an application or business process from a hypothetical scenario and create a migration plan for moving it to the cloud. Define the migration steps, assess potential risks, and create a timeline for the migration process.
11. Data Center Design: Design a hypothetical data center based on given requirements. Determine the physical area, power and cooling requirements, network bandwidth, and select a geographic location considering factors like natural hazards and availability of local technical talent.
12. Data Center Reliability and Tier Standard: Study different tier standards for data centers (e.g., Tier I, Tier II, Tier III, Tier IV). Evaluate the reliability and availability requirements for a given scenario and design a data center architecture that meets the desired tier level.

COURSECODE	Principles of Virtualization	Total: 45 hrs Lecture: 15 Practical: 30
CA23CC002		Course Credit: 3-0-2

Course Objectives:

- Provide knowledge in different layers of cloud computing, Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).
- Illustrate the use of various cloud computing technologies.
- Introduce Virtualization technologies: Hypervisor, emulation, and application VM; Platform virtualization, storage virtualization, and network virtualization.
- Provide Introduction to cloud security and secure computation in the cloud.

UNIT	CONTENT	HOURS
1	**Introduction** Introduction to Virtualization - Types of virtualization - Difference between cloud and virtualization - Physical infrastructure and virtual infrastructure – Virtualization approaches - Partitioning - Hosting - Isolation - Hardware independence - Virtual machine - Hypervisor - Types of hypervisor - Virtual machine manager - Types of hypervisor - Introduction to datacenter virtualization Esxi - Difference between Esxi and Esx - Versions of Esxi - Installation and configuration of Esxi 6.0 - vSphere 6.0	9
2	**Components of vSphere 6.0** Components of VMware vSphere - vSphere 6.0: Overview and Architecture - Topology of vSphere 6.0 Data Center - vSphere 6.0 Configuration Maximum sv Center Server - vCenter Server Features - Certificate Management - Alarms and Alerts - Monitoring Features - Template Management - Linked Mode Deployment - Storage Features in vSphere - Shared Storage - Storage Protocols - Datastores - Virtual SAN - Virtual Volumes - Networking Features in vSphere - Virtual Networking - Virtual Switches and its types	9
3	**Features of vSphere and NSX** vSphere Resource Management Features - vMotion - Distributed Resource Scheduler (DRS) - - Distributed Power Management (DPM) - Storage vMotion - Storage DRS - Storage I/O Control - Network I/O Control - vSphere Availability Features - vSphere Data Protection - High Availability - Fault Tolerance - vSphere Replication - Introduction to NSX.	9
4	**VSphere Solutions to Data Center Challenges and vSphere Security.** Challenges - Availability Challenges - Scalability Challenges - Management Challenges - Optimization Challenges - Application Upgrade Challenges - Cloud Challenges - Security - Describe the features and benefits of VMware Platform Services Controller - Configure ESXi host access and authorization - Secure ESXi - vCenter Server - and virtual machines - Upgrade ESXi and vCenter Server instance.	9
5	**Resource optimization and resource management** Network Optimization - Configure and manage vSphere distributed switches - Migrate virtual machines from standard switches to distributed switches - Explain distributed switch features such as port mirroring - LACP - QoS tagging - and NetFlow - CPU Optimization - Explain the CPU scheduler operation - NUMA support - and other features that affect CPU performance - Monitor key CPU performance metrics - Memory Optimization - Explain ballooning - memory compression - and host swapping techniques for memory reclamation when memory is overcommitted - Monitor key memory performance metrics - Storage Optimization - Diagnose storage access problems - Configure VMware vSphere Flash Read Cache - Monitor key storage performance metrics.	9

Course Outcomes (as per Bloom's Taxonomy)

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

CO 1	Understand cloud security and secure computation in the cloud
CO 2	Summarize the need of virtualization and cloud.
CO 3	Explain the cloud security protocols and articulate the cloud security matrix.
CO 4	Organize the cloud security services and policies.
CO 5	Analyze the security of guest and host on cloud.
Text Books	• Virtualization Essentials Paperback – 26 Apr 2012 by Matthew Portnoy - wiley publications • VMware Cookbook Paperback – 17 Jul 2012 by Troy - Shroff/O'Reilly; Second edition (17 July 2012)

Reference Books	• Mastering VMware vSphere 5.5 (SYBEX) Paperback – 2014 by Scott Lowe, Nick Marshall, Forbes Guthrie, Matt Liebowitz, Josh Atwell - Wiley (2014) edition.

S.NO	Name of Lab Exercise		
1	Installing and configuring ESXi 5.5/6.0 Server [On Premise]		
2	HOL-1810-01-SDC	Virtualization 101	Introduction to Management with vCenter Server Introduction to vSphere Networking and Security
3	HOL-1810-01-SDC	Virtualization 101	Introduction to vSphere Storage
4	HOL-1808-01-HCI - vSAN v6.6 - Getting Started	vSAN v6.6 - Getting Started	vSAN 6.6 Setup and Enablement vSAN Scale Out with Configuration Assist vSAN All Flash Capabilities
5	HOL-1808-01-HCI - vSAN v6.6 - Getting Started	vSAN v6.6 - Getting Started	vSAN iSCSI Target vSAN Encryption
6	HOL-1808-01-HCI - vSAN v6.6 - Getting Started	vSAN v6.6 - Getting Started	vSAN PowerCLI and ESXCLI vSAN Stretched Cluster
7	HOL-1803-01-NET - VMware NSX - Getting Started	VMware NSX	NSX Manager Installation and Configuration Logical Switching
8	HOL-1803-01-NET - VMware NSX - Getting Started	VMware NSX	Logical Routing Edge Services Gateway
9	HOL-1811-04-SDC - vSphere Security - Getting Started	vSphere Security	Automating Password Complexity for ESXi Users Forensic Security with vRealize Log Insight
10	HOL-1811-04-SDC - vSphere Security - Getting Started	vSphere Security	VM Encryption and Encrypted vMotion Secure Boot for Hosts and VMs No-Cryptography Administrator Roles and Permissions

SANJEEV AGRAWAL GLOBAL EDUCATIONAL UNIVERSITY, BHOPAL

Syllabus

for

Bachelor of Computer Application (BCA)

Data Science (Major Specialization)

Batch 2023
(Under NEP 2020)

School of Computer Application

wef 2023-24

COURSE CODE	INTRODUCTION TO DATA SCIENCE AND ARTIFICIAL INTELLIGENCE	Total Lecture: 60 Theory:45 Tutorial:15
CA23DS001		(LTP=3-0-2)

Course Objectives

- To introduce students to the field of Artificial Intelligence (AI) and its history, including the elements of intelligence, reasoning, cognition, philosophy, math, linguistics, control theory, and computer science in AI.
- To teach students the various subareas of AI, such as fundamental AI methods and algorithms, cognitive computing, neural networks, knowledge-based systems, natural language processing, speech recognition, intelligent agents, robotics, computer vision, machine learning, and deep learning.
- To enable students to comprehend the Data science (DS) process, including understanding business needs, collecting and exploring data, preparing and processing data, building and validating models, and deploying and monitoring performance.
- To help students examine the applications of AI and data science in various industries, including healthcare, finance, education, military and cyber-security, telecommunications, public safety, media, genetics and genomics, gaming, security, transportation, search engines, and recommendation systems.

UNIT	CONTENTS	HOURS
I	**Module 1: Introduction to Artificial Intelligence (AI)** What is Artificial Intelligence (AI)? Brief history of AI. Intelligence and artificial intelligence. Elements of Intelligence – Reasoning, Learning, Problem Solving, Perception, Linguistic Intelligence. Coming together of cognition, philosophy, math, linguistics, control theory and computer science. Machines and intelligence. The Turing tests. Limitations and possibilities of AI. Concerns about AI. AI and the future.	7
II	**Module 2: Introduction to Data Science (DS)** Definition of data science. What do data scientists do? Why Data Science is Important? Data, overview of the types of data and its scale. Big data. Mathematics, probability and statistics behind data science. Comparison of Data science with Data analytics, Comparison of Data science with Artificial Intelligence.	9
III	**Module 3: Overview of the Data Science Process** Understand the business needs, Collect and explore the data, Prepare and process the data, Overview of Machine learning and Modelling techniques for data science, Build and validate the models, Deploy and monitor the performance.	9
IV	**Module 4: Subareas of AI:** Intelligent Agents, Problem Solving Strategies, Cognitive Computing, Knowledge-based systems, Neural Networks, Machine Learning, Deep Learning, Natural Language Processing, Speech Recognition, Robotics, Computer Visio.	10
V	**Module 5: Programming Languages and Tools for Data Science Introduction to Data Science** - Why Python? Installing Python: Python, PyCharm, Anaconda, VS Code, Jupyter Notebooks, and Colab. Essential Python libraries, Python Introduction- Features, Identifiers, Reserved words, Indentation, Comments, Built-in Data types and their Methods: Strings, List, Tuples, Dictionary, Set Type Conversion, Operators. **Decision Making**- Looping, Loop Control statement, Math and Random number functions. User defined functions, function arguments and its types.	10

Course Outcomes as per Bloom's Taxonomy

At the end of the course the students should be able to:

CO1	Discuss artificial intelligence, its history, the various subareas of AI and their applications, and its limitations.
CO2	Compare and contrast data science, data analytics, and AI, and understand the mathematics and statistics behind data science.
CO3	Explain the data science process and the modelling techniques used in data science.
CO4	Identify potential opportunities and challenges of AI and data science in various industries.
CO5	Develop an understanding of the current state of AI and its future possibilities.
Text Book	• John W. Foreman, Python for Data Science for Dummies - Luca Massaron and John Paul Mueller, John Wiley & Sons, Inc. Wiley Publication. • Joel Grus, Data Science from Scratch: First Principles with Python, PHI. • Sinan Ozdemir, Principles of Data Science, PACKT

	• Davy Cielen, Aeno D. B. Meysman, Mohamed Ali, Introducing Data Science, Manning • Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, Pearson.

List of Experiments:

1. *Tutorial:* Installing Python: Install Python on your computer and test that it is working correctly by opening the Python interpreter and executing some basic commands. Using an IDE: Install and use an integrated development environment (IDE) such as PyCharm or Visual Studio Code.
2. Write a Python program to print "Hello World!" on the screen.
3. Write a program to perform basic arithmetic operations such as addition, subtraction, multiplication, and division.
4. A store has a list of products and their prices. You need to create a program that allows the user to perform the following tasks:
    a. View the list of products and prices.
    b. Add a new product and its price to the list.
    c. Remove a product and its price from the list.
    d. Update the price of an existing product.
    e. Search for a product by name and display its price.
    f. Create this program using various data types such as strings, integers, floats, Booleans, lists, and dictionaries to implement these functionalities.
5. Write a program that prompts the user to enter a number, and then prints "Positive" if the number is greater than zero, "Negative" if the number is less than zero, and "Zero" if the number is equal to zero.
6. Write a program that prompts the user to enter an 8- or 9-digits' number, and then prints the factorial of that number using a while loop.
7. Write a program that prints the first 10 numbers of the Fibonacci sequence using a for loop.
8. Write a program that prints the first 10 even and 10 odd numbers using a while loop.
9. Write a program that prompts the user to enter two numbers, and then prints the larger number using a conditional expression.
10. Write a program that prompts the user to enter a number, and then prints "Positive" if the number is greater than zero, "Negative" if the number is less than zero, and "Zero" if the number is equal to zero, using a conditional expression.
11. Create a list of random numbers. Using list comprehension, create a new list that contains only the numbers from the original list which are divisible by 3. Convert this list into a tuple. Create a dictionary where the keys are the indices and the values are elements from the tuple.
12. Define a function that accepts a list of numbers as an argument. Inside the function, implement a conditional statement that checks if the list is empty, has only one element, or has more than one element. For each of these conditions, print a different output. If the list has more than one element, sort the list in ascending order.
13. Write a Python script to simulate the game of "FizzBuzz". Loop through the numbers from 1 to 100. If a number is divisible by 3, print "Fizz". If a number is divisible by 5, print "Buzz". If it's divisible by both 3 and 5, print "FizzBuzz". Otherwise, just print the number.

COURSE CODE	PROBABILITY AND STATISTICS	Total Lecture: 60
CA23DS002		Course Credit: 3-0-0

Course Objectives:

- Understand the fundamental concepts of probability theory and its application in various fields.
- Explore random variables and their statistical properties, including expectations, moments, and limit theorems.
- Learn and apply statistical measures to summarize data, including measures of central tendency, dispersion, and graphical representations.
- Understand and apply correlation and regression analysis techniques to examine relationships and make predictions between variables.
- Study different probability distributions, both discrete and continuous, and their characteristics and applications.

UNIT	CONTENTS	HOURS
1	**Introduction to Probability and Statistics Probability:** History, Terminology of probability, Formula for Probability, priori probability, limitations of classical probability, statistical or empirical probability, theory of sets, elements of sets, operations on sets, algebra of sets, axiomatic approach to probability, theorems on probabilities of events, law of probability theory, Introduction to conditional probability and expectations, examples for conditional probability and expectation, properties of conditional random variables, Bayes theorem and Independence, application of Bayes Theorem, **Statistics:** History, Types of statistics, types of data, variables, types of variables based on measurement, based on observation, the difference between cross-sectional and time series data, sample, population, parameter, estimation, estimator, sampling distribution and standard error, application of statistics in different areas.	9
2	**Random Variables and Expectations Hours** Random variables- discrete, continuous and mixed random variables, statistical properties of random variables, Expectation of random variables, expectation of random variables in terms of variance and covariance, jointly distributed random variables, moment generating function, characteristic function, limit theorems related to random variables. Markov inequality, Chebyshev's inequality. Probability function, probability density function, and probability mass function.	8
3	**Summarizing Data Using Statistical Measures** **Measures of central tendency** - Mean: Arithmetic mean, geometric mean, and harmonic mean with its mathematical properties, the mathematical relationship among these different means, median for raw data–odd number and even number, grouped data, a mode for raw data grouped data, properties mean, median and mode and relationship among mean, median and mode**.** **Measures of dispersion** – range, mean deviation, quartile deviation, standard deviation, variance, covariance and its properties, coefficient of variation, skewness, kurtosis. **Graphical representation of data**: Uni-dimensional, bi-dimensional, and multidimensional.	10
4	**Correlation and Regression** Correlation analysis, simple and multiple correlations, coefficient of correlation, properties of the correlation coefficient, the significance of single correlation coefficient, the significance of multiple correlation coefficient, linear model, assumptions of the linear model, estimation of parameters using OLS, regression analysis, properties of regression coefficients, the significance of regression coefficient, multiple linear regression analysis, assumptions, the significance of estimated parameters.	8
5	**Probability Distribution** Introduction to Probability Distribution, Discrete Probability Distribution: Bernoulli Distribution. Binomial Distribution. Poisson Distribution, Hypergeometric Distribution, Continuous Probability Distribution: Normal Distribution, Exponential Distribution, Gamma Distribution.	10

Course Outcomes as per Bloom's Taxonomy

At the end of the course the students should be able to:

CO1	Apply probability theory concepts and techniques to analyze and solve real-world problems involving uncertainty and randomness.
CO2	Analyze and interpret data using appropriate statistical measures and graphical representations to provide meaningful insights.
CO3	Calculate and interpret correlation coefficients and regression parameters to quantify and understand relationships between variables.
CO4	Identify and apply appropriate probability distributions to model and analyze various types of data and events.

Reference Books	• Introduction to probability Models, Ninth Edition – Sheldon M. Ross, Elsevier Publication, Academic Press,UK • Introduction to Probability and Statistics for Engineers and Scientists, Third Edition - Sheldon M. Ross, Elsevier Publication, Academic Press,UK • An introduction to Probability and Statistical Inference – George Roussas, Academic Press.

SANJEEV AGRAWAL GLOBAL EDUCATIONAL UNIVERSITY, BHOPAL

Syllabus

for

Bachelor of Computer Application (BCA)

Cyber Security (Minor Specialization)

Batch 2023
(Under NEP 2020)

School of Computer Application

wef 2023-24

COURSECODE	CYBER SECURITY ATTACKS AND DEFENSE STRATEGIES	Total: 45 hrs Lecture: 15 Practical: 30
CA23CY001		Course Credit: 3-0-2

Course Objectives:

- To introduce students to the various types of cyber threats and their impact on organizations.
- To provide an understanding of social engineering techniques and how they are used to gain unauthorized access to systems and data.
- To enable students to assess and respond to network security incidents and breaches.
- To teach students common vulnerabilities in software applications and how they can be exploited by attackers.
- To make students understand risk management and security controls.

UNIT	CONTENT	HOURS
1	**Cybersecurity Concepts** Overview of Cybersecurity, History of Cybersecurity, Cybercrime and its types, Common techniques and tools used in cybercrime, Cybersecurity goals, Importance of Cybersecurity, Security Terminologies, CIA Triad and compromise threats, Types of cyber hackers/attackers, Motivations for hacking, Challenges and managing Cybersecurity in a dynamic world, The NIST Cybersecurity Framework, current security trends.	8
2	**Social Engineering and Malware** Phishing attacks, Spam, Dumpster diving, Shoulder surfing, Pharming, Tailgating, Identity fraud, Credential harvesting, Impersonation, Watering hole attack, Enumeration, Password attacks, Malware threats-virus and Trojans, Types of Malwares: Virus, Worms, Trojans, Rootkits, Robots, Adware's, Spywares, Ransomware, Zombies etc. and their behavior.	10
3	**Network Attacks** Network attacks: Layer 2 attacks, Layer 3 attacks, IP spoofing, Man-in-the-middle attacks (MitM), DNS spoofing and hijacking, DDoS attacks, Packet Sniffing, Vulnerability analysis, Foot printing and reconnaissance.	7
4	**Web Application Attacks** Application attacks: Privilege escalation, Broken authentication, and session management, Cross-site scripting, Injections attacks, overflow attacks, Error handling, Improper input handling, API attacks, Memory leak, Replay attacks, Clickjacking, Cross-site request forgery (CSRF), Server-side request forgery (SSRF).	8
5	**Risk Management** Risk types, Risk analysis, Risk management process, Risk mitigation strategies - Avoidance, Transference, Mitigation, and Acceptance, Quantitative and qualitative assessments, Business impact analysis, disaster recovery. Threat Modeling Techniques: STRIDE, DREAD, PASTA, and VAST, Mitigating Threats and Reducing Attack Surfaces. **Security Controls:** Security control categories – Managerial, Operational, Technical, Control types - Preventative, Detective, Corrective, Deterrent, Compensating and Physical. Security policies, standards, frameworks and procedures.	11

Course Outcomes (as per Bloom's Taxonomy)

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

CO 1	Discuss the relevance of cybersecurity and related concepts clearly.
CO 2	Analyze and respond to social engineering attacks and malware infections.
CO 3	Simulate and respond to network attacks in a safe and controlled environment.
CO 4	Use techniques for identifying and mitigating vulnerabilities in software applications.
CO 5	Apply threat modeling techniques to identify and mitigate risks to information systems. Identify and evaluate different categories of security controls.
Text Books	Sammons, John, and Michael Cross. The basics of cyber safety: computer and mobile device safety made easy. Elsevier, 2016. HACKING: Social Engineering Attacks, Techniques & Prevention, Alex Wagner. Network Attacks and Exploitation by Mathew Monte, August 2015, Wiley. Information Security Risk Analysis - Thomas R. Peltier, Third Edition, Pub: Auerbach, 2012.

Reference Books	Cybersecurity Essentials, Brooks, Charles J., Christopher Grow, Philip Craig, and Donald Short. John Wiley & Sons, 2018.

List of Experiments:

Demonstration of social engineering attacks.
Demonstrate malware behavior by executing a malware sample on a VM.
Demonstration of DNS Spoofing attack.
Demonstrate DoS and DDoS attacks.
Demonstration of any of the application attacks.
Demonstrate a Cross-site request forgery attack.
Case study: Demonstration of Risk posture, risk analysis, risk assessment and mitigation of an example organization.
Demonstrate the use of different types of security controls for layered security.

COURSECODE	DATA COMMUNICATION AND COMPUTER NETWORKS	Total: 45 hrs Lecture: 15 Practical: 30
CA23CY002		Course Credit: 3-0-2

Course Objectives:

- Gain a comprehensive understanding of networking fundamentals, including network types, architectures, topologies, transmission media, and networking models.
- Develop proficiency in application layer protocols and network devices, such as DHCP, DNS, HTTP/HTTPS, FTP, Telnet, NIC, hub, switch, bridge, router, and gateways.
- Acquire a strong grasp of session layer, transport layer, and network layer concepts, including protocols (TCP, UDP, IP, ICMP), IP addressing, subnetting, routing, and switching.
- Master data link layer and physical layer concepts, including framing techniques, error detection and correction, MAC protocols, Ethernet standards, wireless networking, transmission media, and signal encoding.
- Explore emerging networking technologies, including WAN technologies, internet connectivity options, remote access, network security, network operating systems, troubleshooting tools, and system monitoring tools.

UNIT	CONTENT	HOURS
1	**Communication and Networking Fundamentals** Basic Concepts: Components of data communication, distributed processing, standards and organizations, Basics of Network and Networking, Advantages of Networking, Types of Networks, Network Terms- Host, Workstations, Server, Client, Node, Types of Network Architecture- Peer-to-Peer and Client/Server, Workgroup Vs. Domain. Network Topologies, Types of Topologies, Logical and physical topologies, selecting the Right Topology, Types of Transmission Media, Communication Modes, Wiring Standards and Cabling- straight through cable, crossover cable, rollover cable, media connectors (Fiber optic, Coaxial, and TP etc.) Introduction of OSI model, Seven layers of OSI model, Functions of the seven layers, Introduction of TCP/IP Model, TCP, UDP, IP, ICMP, ARP/RARP, Comparison between OSI model & TCP/IP model. Overview of Ethernet Addresses.	9
2	**Application Layer, Presentation Layer and Network Devices** Application Layer: DHCP, DNS, HTTP/HTTPS, FTP, TFTP, SFTP, Telnet, Email: SMTP, POP3/IMAP, NTP. Presentation Layer: Role and functions of Presentation Layer, Data representation and encoding techniques (ASCII, EBCDIC, etc.), Data compression and decompression, Data encryption and decryption, Common protocols at the Presentation Layer (JPEG, MPEG, SSL/TLS, etc.) Networks Devices- NIC-functions of NIC, installing NIC, Hub, Switch, Bridge, Router, Gateways, And Other Networking Devices, Repeater, CSU/DSU, and modem	9
3	**Session Layer, Transport Layer and Network Layer** **Session Layer:** Role and functions of Session Layer, Establishing, managing, and terminating sessions, Session synchronization and recovery, Session layer protocols (NetBIOS, RPC, etc.), Error handling and flow control at the Session Layer. **Transport Layer:** Transmission Control Protocol (TCP), User Datagram Protocol (UDP), Overview of Ports and Sockets. **Network Layer**: Internet Protocol (IP), IP standards, versions, functions, IPv4 addressing, IPv4 address Classes, IPv4 address types, Subnet Mask, Default Gateway, Public & Private IP Address, methods of assigning IP address, IPv6 address, types, assignment, Data encapsulation, The IPv4 Datagram Format, The IPv6 Datagram Format, Internet Control Message Protocol (ICMP), ICMPv4, ICMPv6, Internet Group Management Protocol (IGMP, Introduction to Routing and Switching concepts.	9
4	**Data Link Layer and Physical Layer:** **Data Link Layer:** Role and functions of Data Link Layer, Framing techniques (bit stuffing, byte stuffing, etc.), Error detection and correction (parity, CRC, etc.), MAC addresses and MAC protocols (Ethernet, Wi-Fi, etc.), Switching techniques (store-and-forward, cut-through, etc.). Address Resolution Protocol (ARP) and Reverse ARP (RARP), Error handling and flow control at the Data Link Layer. Wireless Networking: Wireless Technology, Benefits of Wireless Technology, Types of Wireless Networks: Ad-hoc mode, Infrastructure mode, Wireless network Components: Wireless Access Points, Wireless NICs, wireless LAN standards: IEEE 802.11a, IEEE 802.11b, IEEE 802.11g, wireless LAN modulation techniques, wireless security Protocols: WEP, WPA, 802.1X, Installing a wireless LAN. **Physical Layer**: Role and functions of Physical Layer, Transmission media (twisted pair, coaxial, fiber optics, etc.), Signal encoding and modulation techniques, Multiplexing and demultiplexing, Physical layer protocols and standards (RS-232, V.35, etc.), Wireless communication technologies (Wi-Fi, Bluetooth,	9

	etc.), Wireless network architectures and protocols, Line coding and scrambling techniques, Analog and digital transmission, Signal quality and noise considerations.	
5	**Other Networking technologies** What Is a WAN?, WAN Switching, WAN Switching techniques Circuit Switching, Packet Switching etc., Connecting to the Internet : PSTN, ISDN, DSL, CATV, Satellite-Based Services, Last Mile Fiber, Cellular Technologies, Connecting LANs : Leased Lines, SONET/SDH, Packet Switching, Remote Access: Dial-up Remote Access, Virtual Private Networking, SSL VPN, Remote Terminal Emulation, Network security: Authentication and Authorization, Tunneling and Encryption Protocols, IPSec, SSL and TLS, Firewall, Other Security Appliances, Security Threats. **Network Operating Systems:** Microsoft Operating Systems, Novell NetWare, UNIX and Linux Operating Systems, Macintosh Networking, Trouble Shooting Networks: Command-Line interface Tools, Using Network Utilities: ping, traceroute, tracert, ipconfig, arp, nslookup, netstat, nbtstat, Hardware trouble shooting tools, system monitoring tools. Emerging networking technologies.	**9**

Course Outcomes (as per Bloom's Taxonomy)	
At the end of the course the students will be able to:	
CO 1	Discuss the fundamental principles and components of networking, enabling effective communication and collaboration within various network environments.
CO 2	Demonstrate proficiency in configuring, managing network devices, protocols, and services to establish reliable and secure network connections
CO 3	Identify the knowledge of network layers and protocols to ensure efficient data transmission, address allocation, and routing within interconnected networks.
CO 4	Implement data link and physical layer technologies, including error detection, media selection, and wireless networking, to establish and maintain robust network connections.
CO 5	Describe the emerging networking technologies, enabling adaptation to evolving network infrastructure and effective troubleshooting of network issues.
Text Books	• "Computer Networks: A Systems Approach" by Morgan Kaufmann (Publication date: 2019) • "Computer Networks: A Top-Down Approach" by James F. Kurose and Keith W. Ross (Publication date: 2016)
ReferenceBooks	• "Data Communications and Networking" by Behrouz A. Forouzan (Publication date: 2013) latest publication. • "Computer Networks: A Systems Approach" by Larry L. Peterson and Bruce S. Davie (Publication date: 2019). • "Data and Computer Communications" by William Stallings (Publication date: 2017)

List of Experiments:

Switch Configuration - Basic Commands and Switch Port Security.

Using Packet Tracer or a similar network simulation tool, configure a switch with the following requirements:

- Set a unique hostname for the switch.
- Assign an IP address to the switch and enable SSH for remote management.
- Implement switch port security on a specific interface to allow only two MAC addresses and restrict any additional connections.
- Specify a violation action that shuts down the port if an unauthorized device is connected.
- Test the switch port security by connecting unauthorized devices and observe the resulting actions taken by the switch.

2. **Router – Configuration and Setting up of Passwords.**

   Configure a router with appropriate passwords for console, Telnet, and privileged mode access, and verify the successful setup of passwords.

3. **PPP Encapsulation, PPP PAP Authentication, PPP CHAP Authentication.**

   Set up a PPP connection between two routers using PPP encapsulation and configure both PAP and CHAP authentication protocols to establish secure communication. Test the authentication process and verify the successful establishment of the PPP connection.

4. **A configuration of default, Static and Dynamic Routing.**

   Configure a network with three routers and implement default routing, static routing, and dynamic routing using a routing protocol of your choice. Test the routing configurations by verifying proper forwarding of packets between different networks.

5. **VLAN Configuration.**

   Design a network with multiple VLANs and configure VLANs on switches, along with inter-VLAN routing, to enable communication between different VLANs.

6. **Configuration of Access-lists - Standard and Extended ACLs.**
   Configure standard and extended access-lists on a router to control traffic flow based on specific source/destination IP addresses, protocols, and ports. Test the access-lists by allowing or denying specific traffic patterns and verify the desired filtering behavior.
7. **DHCP, DHCP Relay and DHCP Exclusions.**
   Configure a network with a DHCP server, DHCP relay agent, and multiple client devices, and set up DHCP exclusions to reserve specific IP addresses. Test the DHCP functionality by observing IP address assignment and verify the DHCP relay agent's role in forwarding DHCP messages between networks.
8. **Configuring Logging to a Remote Syslog Server.**
   Configure a router to send logging messages to a remote syslog server and analyze the received logs to identify and troubleshoot network events or issues.
9. **Analyze network with a router**
   Design and analyze network with a router, Switch and Hub to find the number of broadcast domains and collision domain using packet tracer.
10. **Configure a wireless network for ad-hoc and infrastructure mode.**
    Configure a wireless network using two wireless devices to operate in both ad-hoc and infrastructure modes, and test the connectivity and functionality of each mode.
11. **Configure point to site and site to site VPN.**
    Configure a point-to-site and site-to-site VPN connection between two routers/firewalls, and verify secure communication between a remote client and the network, as well as between two separate networks.
12. **Troubleshooting**
    Perform network troubleshooting using ping, traceroute, tracert, ipconfig, arp, nslookup, netstat, nbtstat.

SANJEEV AGRAWAL GLOBAL EDUCATIONAL UNIVERSITY, BHOPAL

Syllabus

for

Bachelor of Computer Application (BCA)

Artificial Intelligence (Minor Specialization)

Batch 2023
(Under NEP 2020)

School of Computer Application

wef 2023-24

COURSE CODE	KNOWLEDGE REPRESENTATION AND REASONING	Total Lecture: 60 Theory:45 Tutorial:15
CA23AI001		(LTP=3-1-0)

Course Objectives

- Enhancing the basic understanding of intelligent agents and their role in problem-solving and decision-making.
- Describe different problem-solving techniques and search algorithms to find optimal solutions.
- Explain about the gaming concept in AI.
- Develop knowledge and reasoning skills to represent, manipulate, and reason with knowledge.
- Explore different learning approaches and techniques to build adaptive and intelligent agents.

UNIT	CONTENTS	HOURS
I	**Intelligent System and Intelligent Agent** What is intelligence? Structure of intelligent system, biological brain and Basic neural model. Intelligent Agents, How Agents Should Act, Structure of Intelligent Agents, Simple reflex agents, Goal-based agents, Utility-based agents, Environments, Environment programs.	8
II	**Problem Solving by Searching** Problem-Solving Agents, Toy problems, Real-world problems, Searching for Solutions, Uniformed Search Strategies, Informed Search Strategies, Heuristic Functions. **Beyond Classical Search** Local Search Algorithms and Optimization Problems, Local Search in Continuous Spaces, Searching with Nondeterministic Actions, Searching with Partial Observations, Online Search Agents and Unknown Environments.	14
III	**Adversarial Search** Games, Optimal Decisions in Games, Alpha-Beta Pruning, Imperfect Real-Time Decisions, Stochastic Games, Partially Observable Games, State-of-the-Art Game Programs, Alternative Approaches.	12
IV	**Constraint Satisfaction Problems** Introduction to Constraint Satisfaction Problems, Constraint Propagation, Backtracking Search for Constraint Satisfaction Problems, Local Search for Constraint Satisfaction Problems, The Structure of Problems.	10
V	**Knowledge and Reasoning** **Logical Agents:** Knowledge-Based Agents, The Wumpus World, Logic, Propositional Logic, Propositional Theorem Proving, Effective Propositional Model Checking, Agents Based on Propositional Logic. **First-Order Logic:** Syntax and Semantics, Extensions and Notational Variations, Using First-Order Logic. **Inference in First-Order Logic:** Proposition vs First-Order Inference, Unification and Lifting, Forward Chaining, Backward Chaining, Resolution.	16

Course Outcomes as per Bloom's Taxonomy

At the end of the course the students should be able to:

CO1	Understand the foundations, history and the role of intelligent agents in solving complex problems.
CO2	Analyze and compare different search algorithms to find optimal solutions for well-defined problems.
CO3	Design and implement knowledge-based systems using logical and rule-based representations.
CO4	Develop planning agents capable of generating and executing plans in complex environments.
CO5	Understand the concept of gaming and know the decision making in checker, go, etc. games.
Text Book	• Artificial Intelligence: A Modern Approach, Stuart J. Russell and Peter Norvig, Second Edition, Pearson Education, Inc., Upper Saddle River; New Jersey 07458. • Artificial Intelligence: The Basics by Kevin Warwick, Professor of Cybernetics Kevin, 1st published 2012 by Routledge. • "Artificial intelligence A systems approach" by M. Tim Jones, INFINITY SCIENCE PRESS LLC, 2008.

Reference Books	• D. Poole, Artificial Intelligence: Foundations of Computational Agents, Cambridge University Press, 2010. • "Artificial Intelligence and Intelligent Systems", by Padhy N.P, 4th impression, Oxford University Press, 2007. • Super intelligence paths, Dangers and Strategies by Nick Bostrom, Oxford University Press.
List of Experiments: 1.Implementing a simple reflex agent for a basic task, such as vacuum cleaning or navigation. 2.Designing and implementing a search algorithm to solve a toy problem, such as the eight-puzzle problem.	

COURSE CODE	COMPUTER NETWORKS	Total Lecture: 60 Theory:45 Tutorial:15
CA23AI002		(LTP=3-0-2)

Course Objectives:

- Gain a comprehensive understanding of networking fundamentals, including network types, architectures, topologies, transmission media, and networking models.
- Develop proficiency in application layer protocols and network devices, such as DHCP, DNS, HTTP/HTTPS, FTP, Telnet, NIC, hub, switch, bridge, router, and gateways.
- Acquire a strong grasp of session layer, transport layer, and network layer concepts, including protocols (TCP, UDP, IP, ICMP), IP addressing, subnetting, routing, and switching.
- Master data link layer and physical layer concepts, including framing techniques, error detection and correction, MAC protocols, Ethernet standards, wireless networking, transmission media, and signal encoding.
- Explore emerging networking technologies, including WAN technologies, internet connectivity options, remote access, network security, network operating systems, troubleshooting tools, and system monitoring tools.

UNIT	CONTENTS	HOURS
I	**Networking Fundamentals** Basics of Network and Networking, Advantages of Networking, Types of Networks, Network Terms- Host, Workstations, Server, Client, Node, Types of Network Architecture- Peer-to-Peer and Client/Server, Workgroup Vs. Domain. Network Topologies, Types of Topologies, Logical and physical topologies, selecting the Right Topology, Types of Transmission Media, Communication Modes, Wiring Standards and Cabling- straight through cable, crossover cable, rollover cable, media connectors (Fiber optic, Coaxial, and TP etc.) Introduction of OSI model, Seven layers of OSI model, Functions of the seven layers, Introduction of TCP/IP Model, TCP, UDP, IP, ICMP, ARP/RARP, Comparison between OSI model & TCP/IP model. Overview of Ethernet Addresses and Independence, Linear Transformations. Eigen values, Eigen Vectors, Cayley – Hamilton Theorem.	9
II	**Application Layer, Presentation Layer and Network Devices** Application Layer: DHCP, DNS, HTTP/HTTPS, FTP, TFTP, SFTP, Telnet, Email: SMTP, POP3/IMAP, NTP. Presentation Layer: Role and functions of Presentation Layer, Data representation and encoding techniques (ASCII, EBCDIC, etc.), Data compression and decompression, Data encryption and decryption, Common protocols at the Presentation Layer (JPEG, MPEG, SSL/TLS, etc.) Network Devices- NIC- functions of NIC, installing NIC, Hub, Switch, Bridge, Router, Gateways, And Other Networking Devices, Repeater, CSU/DSU, and modem	9
III	**Session Layer, Transport Layer and Network Layer** **Session Layer:** Role and functions of Session Layer, Establishing, managing, and terminating sessions, Session synchronization and recovery, Session layer protocols (NetBIOS, RPC, etc.), Error handling and flow control at the Session Layer. **Transport Layer:** Transmission Control Protocol (TCP), User Datagram Protocol (UDP), Overview of Ports and Sockets. **Network Layer**: Internet Protocol (IP), IP standards, versions, functions, IPv4 addressing, IPv4 address Classes, IPv4 address types, Subnet Mask, Default Gateway, Public & Private IP Address, methods of assigning IP address, IPv6 address, types, assignment, Data encapsulation, The IPv4 Datagram Format, The IPv6 Datagram Format, Internet Control Message Protocol (ICMP), ICMPv4, ICMPv6, Internet Group Management Protocol (IGMP), Introduction to Routing and Switching concepts.	9
IV	**Data Link Layer and Physical Layer:** **Data Link Layer:** Role and functions of Data Link Layer, Framing techniques (bit stuffing, byte stuffing, etc.), Error detection and correction (parity, CRC, etc.), MAC addresses and MAC protocols (Ethernet, Wi-Fi, etc.), Switching techniques (store-and-forward, cut-through, etc.). Address Resolution Protocol (ARP) and Reverse ARP (RARP), Error handling and flow control at the Data Link Layer. Wireless Networking: Wireless Technology, Benefits of Wireless Technology, Types of Wireless Networks: Ad-hoc mode, Infrastructure mode, Wireless network Components: Wireless Access Points, Wireless NICs, wireless LAN standards: IEEE 802.11a, IEEE 802.11b, IEEE 802.11g, wireless LAN modulation techniques, wireless security Protocols: WEP, WPA, 802.1X, Installing a	9

	wireless LAN. **Physical Layer**: Role and functions of Physical Layer, Transmission media (twisted pair, coaxial, fiber optics, etc.), Signal encoding and modulation techniques, Multiplexing and demultiplexing, Physical layer protocols and standards (RS-232, V.35, etc.), Wireless communication technologies (Wi-Fi, Bluetooth, etc.), Wireless network architectures and protocols, Line coding and scrambling techniques, Analog and digital transmission, Signal quality and noise considerations.	
V	**Other Networking technologies** What Is a WAN?, WAN Switching, WAN Switching techniques Circuit Switching, Packet Switching etc., Connecting to the Internet : PSTN, ISDN, DSL, CATV, Satellite-Based Services, Last Mile Fiber, Cellular Technologies, Connecting LANs : Leased Lines, SONET/SDH, Packet Switching, Remote Access: Dial-up Remote Access, Virtual Private Networking, SSL VPN, Remote Terminal Emulation, Network security: Authentication and Authorization, Tunneling and Encryption Protocols, IPSec, SSL and TLS, Firewall, Other Security Appliances, Security Threats. **Network Operating Systems:** Microsoft Operating Systems, Novell NetWare, UNIX and Linux Operating Systems, Macintosh Networking, Trouble Shooting Networks: Command-Line interface Tools, Using Network Utilities: ping, traceroute, tracert, ipconfig, arp, nslookup, netstat, nbtstat, Hardware trouble shooting tools, system monitoring tools. Emerging networking technologies. modulation techniques, Multiplexing and demultiplexing, Physical layer protocols and standards (RS-232, V.35, etc.), Wireless communication technologies (Wi-Fi, Bluetooth, etc.), Wireless network architectures and protocols, Line coding and scrambling techniques, Analog and digital transmission, Signal quality and noise considerations.	**9**

Course Outcomes as per Bloom's Taxonomy	
At the end of the course the students should be able to:	
CO1	Discuss the fundamental principles and components of networking, enabling effective communication and collaboration within various network environments.
CO2	Demonstrate proficiency in configuring, managing network devices, protocols, and services to establish reliable and secure network connections.
CO3	Identify the knowledge of network layers and protocols to ensure efficient data transmission, address allocation, and routing within interconnected networks.
CO4	Implement data link and physical layer technologies, including error detection, media selection, and wireless networking, to establish and maintain robust network connections.
CO5	Describe the emerging networking technologies, enabling adaptation to evolving network infrastructure and effective troubleshooting of network issues.
Text Book	• "Computer Networks: A Systems Approach" by Morgan Kaufmann (Publication date: 2019) • "Computer Networks: A Top-Down Approach" by James F. Kurose and Keith W. Ross (Publication date: 2016)
Reference Books	• "Data Communications and Networking" by Behrouz A. Forouzan (Publication date: 2013) latest publication. • "Computer Networks: A Systems Approach" by Larry L. Peterson and Bruce S. Davie (Publication date: 2019). • "Data and Computer Communications" by William Stallings (Publication date: 2017)

Practical's List

1. Switch Configuration - Basic Commands and Switch Port Security.
Using Packet Tracer or a similar network simulation tool, configure a switch with the following requirements:
Set a unique hostname for the switch.
Assign an IP address to the switch and enable SSH for remote management.
Implement switch port security on a specific interface to allow only two MAC addresses and restrict any additional connections.
Specify a violation action that shuts down the port if an unauthorized device is connected.
Test the switch port security by connecting unauthorized devices and observe the resulting actions taken by the switch.

2. Router – Configuration and Setting up of Passwords.
Configure a router with appropriate passwords for console, Telnet, and privileged mode access, and verify the successful setup of passwords.

3. PPP Encapsulation, PPP PAP Authentication, PPP CHAP Authentication.
Set up a PPP connection between two routers using PPP encapsulation and configure both PAP and CHAP authentication protocols to establish secure communication. Test the authentication process and verify the successful establishment of the PPP connection.

4. A configuration of default, Static and Dynamic Routing.
Configure a network with three routers and implement default routing, static routing, and dynamic routing using a routing protocol of your choice. Test the routing configurations by verifying proper forwarding of packets between different networks.

5. VLAN Configuration.

Design a network with multiple VLANs and configure VLANs on switches, along with inter-VLAN routing, to enable communication between different VLANs.

6. Configuration of Access-lists - Standard and Extended ACLs.

Configure standard and extended access-lists on a router to control traffic flow based on specific source/destination IP addresses, protocols, and ports. Test the access-lists by allowing or denying specific traffic patterns and verify the desired filtering behavior.

7. DHCP, DHCP Relay and DHCP Exclusions.

Configure a network with a DHCP server, DHCP relay agent, and multiple client devices, and set up DHCP exclusions to reserve specific IP addresses. Test the DHCP functionality by observing IP address assignment and verify the DHCP relay agent's role in forwarding DHCP messages between networks.

8. Configuring Logging to a Remote Syslog Server.

Configure a router to send logging messages to a remote syslog server and analyze the received logs to identify and troubleshoot network events or issues.

9. **Analyze network with a router**

Design and analyze network with a router, Switch and Hub to find the number of broadcast domains and collision domain using packet tracer.

10. **Configure a wireless network for ad-hoc and infrastructure mode.**

Configure a wireless network using two wireless devices to operate in both ad-hoc and infrastructure modes, and test the connectivity and functionality of each mode.

11. Configure point to site and site to site VPN.

Configure a point-to-site and site-to-site VPN connection between two routers/firewalls, and verify secure communication between a remote client and the network, as well as between two separate networks.

12. Troubleshooting

Perform network troubleshooting using ping, traceroute, tracert, ipconfig, arp, nslookup, netstat, nbtstat.

SANJEEV AGRAWAL GLOBAL EDUCATIONAL UNIVERSITY, BHOPAL

Syllabus

for

Bachelor of Computer Application (BCA)

Advanced Programming

(Interdisciplinary Minor Specialization)

Batch 2023
(Under NEP 2020)

School of Computer Application

wef 2023-24

COURSE CODE	PROGRAMMING PRACTICE USING C	Total Lecture: 60 Theory:45 Tutorial:15
AC23AP001		(LTP=3-0-2)

Course Objectives

- Develop a solid understanding of fundamental programming concepts and problem-solving techniques using the C programming language.
- Acquire proficiency in writing well-structured and efficient C programs by utilizing algorithmic thinking and stepwise refinement.
- Familiarize oneself with different data types, operators, and expressions in C, enabling effective manipulation and representation of data.
- Gain hands-on experience in utilizing control structures such as decision-making statements and looping constructs to create flexible and interactive programs.
- Master essential programming skills involving functions, arrays, strings, pointers, file handling, and data structures like structures, unions, and enumerated types in the C language.

UNIT	CONTENTS	HOURS
I	**Programming and C:** Introduction to computers and computer-based problem solving, Algorithm and Flowchart, Top-down design and stepwise refinement, Programming environment –Assemblers, Compilers, Interpreters. Introduction to Structure of a C program, preprocessor directives, Compiling and executing C programs.	9
II	**Data Types, Decision Control and Looping Statements:** Data Types, Input/Output Statements in C, Constants, Variables, Scope of Variables, Operators, Expressions. Type conversion, Type casting, Decision control- if, if-then-else, nested if, nested else. Looping statements- while, Do-While, for, switch, break, continue and go to statements. Type modifiers and storage class specifiers for data types.	9
III	**Functions:** Introduction to functions, function definition, function declaration, function call, return statement, passing parameters to functions: Call by Value, Call by reference, recursive functions. Arrays: Declaration of arrays, accessing the elements of an array, storing values in arrays, Operations on arrays, Passing arrays to functions, two dimensional arrays, operations on two-dimensional arrays, multidimensional arrays, applications of arrays.	9
IV	**Strings and Pointers:** Strings-Introduction, string taxonomy, operations on strings, Miscellaneous string and character functions, arrays of strings. Pointers- The & and * operator, pointer expression, initializing pointers, malloc vs calloc, array of pointers, pointers to pointers, pointers to functions, function returning pointers.	9
V	**Files, Structures, Unions and Enumerated data types:** File Handling – Files: Introduction to files, using files in C, reading and writing data files, detecting end of file .The file pointer, file accessing functions, fopen, fclose, putc, getc, fprintf. Structure, Union, and Enumerated Data Type: Introduction, structures and functions, Unions, unions inside structures, Enumerated data type.	9

Course Outcomes as per Bloom's Taxonomy

At the end of the course the students should be able to:

CO1	Understand fundamental concepts and principles of programming using the C language.
CO2	Implement decision control statements, such as if, if-else, nested if, and nested else, to control program flow based on specific conditions.
CO3	Use functions, arrays in program development.
CO4	Utilize strings and pointers effectively in C programming.
CO5	Apply file handling techniques and utilize structures, unions, and enumerated data types in C programs.
Text Book	The C Complete Reference by Herbert Schildt - 4th edition, McGraw Hill Education, July 2017. "C Programming: The Complete Guide for Beginners to Master C Programming Step by Step" by Byron Francis, published by Independently published in 2021.

Reference Books	Programming in ANSI C by Balaguruswamy, 3rd Edition, 2005, Tata McGraw Hill. Let us C by Yashwant Kanetkar, 6th Edition, PBP Publication The C programming Language by Richie and Kenninghan, 2004, BPB Publication.

List of Experiments:

1.Write a Program to calculate and display the volume of a CUBE having its height (h=10cm), width (w=12cm) and depth (d=8cm).

2.Write a program to take input of name, roll no and marks obtained by a student in 4 subjects of 100 marks each and display the name, roll no with percentage score secured.

3.Write a program to print whether a given number is even or odd.

4. Write a program to print positive integers from 1 to 10.

5. Write a program to insert 5 elements into an array and print the elements of the array.

6.Write a program to demonstrate the functions and its types.

7. Write a program to calculate factorial of a number using recursion.

8. Write a program to find the biggest among three numbers using pointer.

9. Write a C program to create, declare and initialize structure.

10.Write a program to create a file called emp.rec and store information about a person, in terms of his name, age and salary.

11.Write a C program to add two matrices A and B of size 3x3 and store the result in matrix C.

12. Check whether the given string is a palindrome or not.

COURSE CODE	OPERATING SYSTEM	Total Lecture: 60 Theory:45 Tutorial:15
AC23AP002		Course Credit: 3-0-2

Prerequisites: Computer Fundamentals and Programming Knowledge

Course Objectives:

- Understand the core concepts and principles of operating systems, including their structure, design, and implementation, in order to analyze and evaluate different operating systems effectively.
- Gain proficiency in process management, including process scheduling, multi-threading, and inter-process communication, to ensure efficient and concurrent execution of tasks within an operating system.
- Develop a comprehensive understanding of memory management techniques, including memory allocation, paging, segmentation, and virtual memory, to optimize memory utilization and enhance system performance.
- Acquire the skills to design and implement file systems, including file organization, access methods, and file sharing, while ensuring data protection and efficient storage management.
- Explore input/output systems, including I/O devices, device drivers, I/O operations, and control mechanisms, to optimize I/O performance, manage device resources, handle errors, and understand the significance of I/O virtualization in virtualized environments.

UNIT	CONTENTS	HOURS
I	Introduction to Operating Systems Computer System Organization, Computer System architecture, Operating System generations, Operating System structure, Operating System Types, Operating System design and implementation, Operating System Operations-Process management, Memory management, Storage management, Protection and security, Special-purpose systems, Operating System Services-Process scheduling, Memory allocation, Interrupt handling, Device driver management, File system operations, Network protocol handling, Error handling, Power management, User interface, System calls, System programs, Computing environments, System boot. Case Study on Windows/ Linux.	9
II	**Process Management** Process concept, Process scheduling- Scheduling criteria, Scheduling algorithms, Multiple-Processor scheduling, Operations on processes, Inter-process communication, Thread scheduling- Multi-Threaded Programming Overview, Multithreading models, Thread Libraries, Threading issues, Case Study on Windows/ Linux.	9
III	**Process Synchronization and Deadlocks** Process Synchronization, The Critical section problem, Peterson's solution, Synchronization hardware, Semaphores, Classical problems of synchronization, Monitors, Deadlocks System model, Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection and recovery from deadlock- Case Study on Windows/ Linux.	8
IV	**Memory Management and File System** Memory Management Background, Strategies, Swapping, Contiguous memory allocation, Paging, Structure of page table, Segmentation, Virtual Memory Management Background, Demand paging, Copy-on-write, Page replacement, Allocation of frames, Thrashing. Case Study on Windows/ Linux. File System: File concept, Access methods, Directory structure, File system mounting, File sharing, Protection. Implementing File System: File system structure, File system implementation, Directory implementation, Allocation methods, Free space management- Case Study on Windows/ Linux.	10
V	**Input-Output Systems** Introduction to I/O Systems: Overview of I/O devices and their characteristics, the role of I/O systems in the overall system architecture. I/O Operations and I/O Control: Study of I/O operations, including reading from and writing to I/O devices. I/O Scheduling: Study of popular I/O scheduling algorithms, such as FCFS (First-Come-First-Served), SSTF (Shortest Seek Time First), SCAN, C-SCAN, etc. Buffering and Caching: Introduction to the concepts of buffering and caching in I/O systems. Device Management: Study of device management techniques, including device allocation, device reservation, and dealing with device conflicts. Error Handling and Recovery: Overview of error handling and recovery mechanisms in I/O systems. I/O Virtualization: Introduction to I/O virtualization techniques and their significance in virtualized environments. Understanding how virtualization allows multiple virtual machines to share physical I/O devices efficiently.	10

Course Outcomes as per Bloom's Taxonomy		
At the end of the course the students should be able to:		
CO1	Discuss the core concepts and principles of operating systems.	
CO2	Evaluate the performance and scalability of process management techniques in different scenarios.	
CO3	Compare different memory management techniques, such as memory allocation strategies (e.g., first-fit, best-fit, worst-fit), paging, segmentation, and virtual memory.	
CO4	Describe the principles of file system mounting and directory structures, and their role in managing file organization and hierarchical file access.	
CO5	Identify different I/O scheduling algorithms, such as FCFS, SSTF, SCAN, and C-SCAN, in terms of their advantages and limitations.	
Text Book	• "Operating System Concepts" by Abraham Silber Schatz, Greg Gagne, and Peter B. Galvin - Tenth edition published in 2018 • "Operating Systems: Principles and Practice" by Thomas Anderson and Michael Dahlin - Second edition published in 2014. • "Windows Internals, Part 2: Covering Windows Server 2016 and Windows 10" by Mark E. Russinovich, David A. Solomon, and Alex Ionescu - Seventh edition published in 2017.	
Reference Books	• "Modern Operating Systems" by Andrew S. Tanenbaum and Herbert Bos - Fourth edition published in 2014. • "Operating Systems: Three Easy Pieces" by Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau - First edition published in 2015.	

List of Experiments:

1. Implement a program that simulates different process scheduling algorithms (e.g., First-Come, First-Served, Round Robin, Priority Scheduling) using a set of processes with arrival times and burst times.
2. Develop a program that simulates memory allocation strategies (e.g., First-Fit, Best-Fit, Worst-Fit) by allocating and deallocating memory blocks for a set of processes with varying memory requirements.
3. Write a program that performs common file system operations like creating a file, reading from a file, writing to a file, and deleting a file using system calls or file system APIs.
4. Implement a program that simulates different disk scheduling algorithms (e.g., FCFS, SSTF, SCAN, C-SCAN) to demonstrate how disk I/O requests are scheduled and serviced.
5. Create a simple shell or command-line interface program that can execute basic commands such as listing files, changing directories, creating files/folders, and executing other programs.
6. Develop a program that demonstrates synchronization mechanisms like semaphores or mutexes to solve the classical synchronization problems like the producer-consumer problem or the dining philosopher's problem.
7. Write a program that detects and recovers from deadlock situations using resource allocation graphs or deadlock detection algorithms like Banker's algorithm.
8. Create a program that visually demonstrates how different CPU scheduling algorithms work by simulating the movement of processes in a CPU scheduling queue and the execution on the CPU.
9. Execute 25 basic commands of UNIX.
10. Develop a simple paging system that includes a page table and a mechanism for translating virtual addresses to physical addresses. Implement page fault handling and page replacement algorithms such as FIFO or LRU.
11. Build a segmentation mechanism that supports variable-sized memory segments. Implement functions for segment creation, deletion, and protection.
12. Implement a basic file system that includes file creation, reading, and writing functionalities. Use a simple data structure to maintain file metadata and allocate storage space for files.