21UB4C2DAP:DESIGN AND ANALYSIS OF ALGORITHM

Total Teaching Hours: 48	No. Of Lecture Hours/Week:03
Max Marks:60	Credits: 03

Course Objectives:

- Analyze the asymptotic performance of algorithms.
- Ability to analyze asymptotic runtime complexity of algorithms including formulating recurrence relations.
- Ability to understand and design algorithms using greedy strategy, divide andconquer approach, dynamic programming
- Demonstrate a familiarity with major algorithms and data structures.

Course Outcome:

At the end of this course student will:
CO1: Use divide-and-conquer techniques for solving suitable problems
CO2: Use greedy approach to solve an appropriate problem for optimal solution.
CO3: Apply dynamic programming approach to solve suitable problems
CO4: Understand the limitations of algorithm power and study how to cope with the limitations of algorithm power for various problems

UNIT 1	**Teaching Hours :12**

Introduction:

Algorithms, Fundamentals of Algorithmic problem solving, important problem types, fundamental data structures. Fundamentals of the Analysis of Algorithm Efficiency; The Analysis Framework, Asymptotic Notation and Basic Efficiency classes, Mathematical Analysis of Non-recursive and Recursive Algorithms, Empirical Analysis of Algorithms.

UNIT 2	**Teaching Hours :12**

Brute Force Method

Brute Force Method: Selection sort and bubble sort, sequential search, Brute-Force string Matching, Exhaustive search, Depth-First Search and Breadth-First Search.
Decrease and Conquer: Insertion sort, topological sorting, Algorithms for generating combinatorial objects, Decrease-by-a-constant-factor Algorithms.
Divide and Conquer: Merge sort, quick sort, binary tree traversals and related properties, strassens matrix multiplication.

UNIT 3	Teaching Hours :12

Space and Time Tradeoffs:

Space and Time Tradeoffs: sorting by counting, Input Enhancement in string matching; Hashing

Dynamic programming: Binomial Coefficient, principle of optimality, optimal binary search trees, Knapsack problem and memory functions, Warshall's and Floyds Algorithms.

Greedy technique: Prims Algorithm, Kruskals Algorithm, Bijkstras Algorithm, Huffman Trees.

UNIT 4	Teaching Hours :12

Limitations of Algorithm Power:

Limitations of Algorithm Power: Lower Bound Arguments, Decision Trees, P, NP and NP Complete Problems. Coping with the Limitations of Algorithm Power

Back Tracking: n queens problem, Hamiltonian Circuit problem, subset-sum problem.

Branch-and-bound: Assignment problem, Knapsack Problem, Traveling Salesman Problem

Text Books:

1. Anany Levitin, " Introduction to the Design and Analysis of Algorithms," 3rd Edition, Pearson 2012.
2. Horowitz, sahni, Rajasekaran, "Fundamentals of Computer Algorithms", 2/e, Universities Press, 2007.

Reference Text Books:

1. Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, Clifford Stein, "Introduction to Algorithms", 3rd Edition, The MIT Press 2009.
2. A. V. Aho J E. Hopcroft, J D Ullmann, "The design and Analysis of Computer Algorithms", Addison Wesley Boston, 1983.
3. Jon Kleinberg, Eva Tardos, "Algorithm Design", Pearson Education, 2006

21UB3C1CNT: COMPUTER NETWORKS

Total Teaching Hours : 52	No. Of Lecture Hours/Week:4
Max Marks:70	
Course Objective:	

- To understand the protocol layering and physical level communication.
- To analyse the performance of a network
- To learn the functions of network layer and the various routing protocols.
- To familiarize the functions and protocols of the Transport layer

Course Outcome :

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

CO1: Explain the importance of data communications and the Internet in supporting business communications and daily activities.

C02: Explain how communication works in data networks and the Internet.

CO3: Analyze the services and features of the various layers of data networks. .

CO4: Analyze the features and operations of various application layer protocols such as Http, DNS, and SMTP.

UNIT 1	Teaching Hours :12

Introduction: Data communications, Components, Data representation ,Data flow
Networks: Network criteria, Network types, Internet history
Network Models: Protocol layering, TCP/IP protocol suite, The OSI model.
Introduction to Physical layer: Transmission impairment, Data rate limits, performance.
Introduction to Data Link Layer :Link Layer Addressing

UNIT 2	Teaching Hours :12

Data link Control :Datalink Layer Protocols, HDLC, Point –To-Point(PPP) ,Media Accesses Control (MAC) , ALOHA, CSMA , CSMA/CD, CSMA/CA, Reservation ,Polling, Token Passing, FDMA,TDMA,CDMA

UNIT 3	Teaching Hours :12

Introduction to the Network Layer: Network layer services, packet switching, network layer performance, IPv4 address.
Network layer protocols: Internet Protocol(IP) , ICMPv4, Mobile IP
Unicast Routing: Routing algorithms, , Unicast routing protocols
Next generation IP: IPv6 addressing .

UNIT 4:	Teaching Hours :12

Introduction to the Transport Layer: Introduction, Transport layer protocols, User datagram protocol, Transmission control protocol, TCP services, TCP Features, Segment, A TCP connection, TCP congestion control, Flow Control, Error Control
Application Layer: World wide web , Electronic mail, Domain name system
Quality of services: Flow Control to implement QoS, Integrated services.

Text Books:

1. Data Communication and Networking Behrouz A. Forouzan Tata McGraw Hill Fifth Edition 2013

Web Reference

1. https://www.geeksforgeeks.org/computer-network-tutorials/
2. https://codescracker.com/networking/index.htm
3. https://www.youtube.com/watch?v=3QhU9jd03a0

Reference Text Books:

1. Computer Networks Andrew Tanenbaum David J Whetherall Printice Hall Fifth Edition 2013
2. Kurose, Ross (2012), Computer Networking: A top down approach, Pearson Education, India 5th Edition

: COMPUTER NETWORKS LAB

1. Execute the following commands
   arp, ipconfig, hostname, netdiag, netstat, nslookup, pathping, ping route, tracert.
2. Study of different types of network cables.
3. Practically implement the cross-wired cable and straight wired cable using crimping tool.
4. Study of network IP address configuration(Classification of address, static and dynamic address)
5. Study of netwok IP address configuration (IPv4 and IPv6, subnet, supernet)
6. Study of network devices (Switch, Router, Bridge)
7. Configure and Connect the computer in LAN
8. Block the website using “Windows Defender Firewall” in windows 10
9. Share the folder in the system and access the files of that folder from other system using IP address.
10. Share the printer in Network, and take the print from OTHER pc.
11. Configuration of WiFi hotspot and connect other devices (mobile/laptop)
12. Configuration of switches.
13. Configuration of I/O box fixing.
14. Making your own patch card.
15. Configuration of VLAN using Packet tracer/GNS3.
16. Configuration of VPN using Packet tracer/GNS3.

References:

1. Paul Browning “101CompTIA Networks, 2018, Reality press ltd.

OPERATING SYSTEM

Total Teaching Hours: 48	No. Of Lecture Hours/Week: 03
Max Marks: 60	Credits : 3
Course Objective:	

- To make the computer system convenient to is in an efficient manner.
- To hide the details of hardware resources from the users.
- To provide users a convenient interface to use the computer system.

Course Outcome:

CO1: Understand the basic computer operating system, interaction among the various components.
CO2: Learn the concepts of file management system.
CO3: Validate the policies for scheduling deadlock, memory management and security.
CO4: Analyze the features of file system, distributed system concepts.

UNIT 1	Teaching hours :12

Introduction: Computer System Organization, Architecture, Structure, Operations, Process Management, Memory Management, Storage Management, Kernel Data Structures, Computing Environments. **Operating System Structures**: Services, System Calls, Types, Operating System Structure, System Boot. Processes: Process Concept, Scheduling, Operations, Interprocess Communication, Multithreaded Programming, Multithreading Models.

UNIT 2:	Teaching Hours :12

Process Synchronization: The Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization Example. **Process Scheduling:** Criteria, Scheduling Algorithms, Multi-Processor Scheduling, Real-time CPU Scheduling. **Deadlocks**: System model, Characterization, Methods for handling deadlocks, Deadlock Prevention, Avoidance, Detection and recovery from deadlock.

UNIT 3:	Teaching Hours :12
Memory Management Strategies: Background, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the page table. **Virtual Memory Management:** Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames, Thrashing, Memory-Mapped files, Allocating Kernel Memory. **File System:** File Concept, Access Methods, Directory and Disk Structure, Protection. **File System Implementation:** Structure.	
UNIT 4:	Teaching Hours 12
File-System and Directory Implementation, Allocation Methods, Free Space Management, Efficiency and Performance, Recovery, Mass-Storage Structure: Overview, Disk Scheduling, Disk Management. Distributed System: Advantages, Types of Network-based OS, Robustness, Design Issues, Distributed File Systems, Case Studies: The Linux System, Windows 10 (Process, Memory, Storage Management)	
Essential Text Book:	
1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: "Operating Systems Concepts", 9th Edition, 2016 India, Wiley.	
Reference Text Books:	
1. *Wiliam Stallings, "Operating Systems- Internals and Design Principles", Pearson, IX Edition, 2018* 2. *D.M Dhamdhere: Operating System- A Concept Based Approach, III edition, Tata McGraw Hill,2015* 3. *Harvey M Deitel, Paul J Deitel, DrChoffines, "Operating System", Pearson Education Limited (Publisher), 3rd Edition.*	

PYTHON PROGRAMMING

Total Teaching Hours : 48	No. Of Lecture Hours/Week:03
Max Marks:60	

Course Objective:

- Learn the syntax and semantics of Python programming language.
- Illustrate the process of structuring the data using lists, tuples, and dictionaries.
- Demonstrate the use of built-in functions to navigate the file system.
- Implement the Object-Oriented Programming concepts in Python.
- Demonstrate the use of Regular Expression and handling file operations

Course Outcome:

CO1: Examine Python syntax and semantics and be fluent in the use of Python flow control and functions.
CO2: Express proficiency in the handling of strings and functions.
CO3: Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples, and sets.
CO4: Identify the commonly used operations involving file systems and regular expressions.
CO5: Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python.

UNIT 1	**Teaching Hours :12**

Introduction to Python Programming Language

Python Interpreter/Shell, Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence, Data types, Indentation, Comments, Reading Input, Print Output, Type Conversions, The type() Function and Is operator. Dynamic and Strongly Typed Language.

Control Flow Statements: Selection, Iteration, Jump statements.

Functions: Built-In Functions, Commonly Used Modules, Function Definition and Calling the Function, The return Statement and void Function. Scope and Lifetime of Variables, Default Parameters.

Strings: Creating and Storing Strings. Basic String operations, Accessing Characters in StringbylndexNumber, String Slicing and Joining, String methods.

UNIT 2	**Teaching Hours :12**

LISTS, Dictionaries, Tuples and Sets

Lists: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, del Statement.

Dictionaries: Creating Dictionary, Accessing and Modifying key: value Pairs in Dictionaries, Built-In Functions Used on Dictionaries, Dictionary Methods, del Statement.

Tuples and Sets: Creating Tuples, Basic Tuple Operations, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists, Relation between Tuples and Dictionaries, Tuple Methods.

UNIT 3	**Teaching Hours :12**

Files and Object-Oriented Programming

Files: Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary Files, Reading and Writing CSV Files.
Object-Oriented Programming: Classes and Objects, Creating Classes in Python, Creating Objects in Python, The Constructor Method, Classes with Multiple Objects, Class Attributes versus Data Attributes, Encapsulation, Inheritance, Polymorphism.

UNIT 4	**Teaching Hours :12**

Data Visualization

Data Visualization: Generating Data-installing Matplotlib, plotting a Simple Line Graph, Random Walks, Rolling Dice with Plotly. Downloading Data- The CSV File Format. Working with dataset and implementation of different graphs.

Text Books:

1. Gowrishankar S, Veena A, "Introduction to Python Programming", 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372

Reference Text Books:

1. Jake VanderPlas, "Python Data Science Handbook: Essential Tools for Working with Data", 1 st Edition, O'Reilly Media, 2016. ISBN-13: 978-1491912058
2. AurelienGeron, "Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems", 1st Edition,O'Reilly Media, 2017. ISBN – 13: 978-1491962299.
3. Wesley J Chun, "Core Python Applications Programming", 3 rd Edition, Pearson Education India, 2015. ISBN-13: 978-9332555365
4. Miguel Grinberg, "Flask Web Development: Developing Web Applications with Python", 2nd Edition, O'Reilly Media, 2018. ISBN-13: 978-1491991732.

21UB4C3ITT INTERNET TECHNOLOGIES

Total Teaching Hours: 48	No. Of Lecture Hours/Week:03
Max Marks:60	Credits: 03
Course Objective:	

- To introduce the fundamentals of Internet, and the principles of web design.
- To construct basic websites using HTML and Cascading Style Sheets.
- To build dynamic web pages with validation using Java Script objects and by applying different event handling mechanisms.
- To develop modern interactive web applications using JSP, Bootstrap ,Angular JS

Course Outcome:

At the end of this course, the students will be able to
CO1: Understand the concepts of World Wide Web.
CO2: Design web pages using the HTML and CSS features with different layouts.
CO3: Development of Web Sites.
CO4: Analysis of Web Sites with dynamic web pages and static web pages.

UNIT 1	**Teaching Hours :12**

INTER Connected NETwork

Internet: The giant wide area network, comminuting over the internet, accessing the internet, internet organization, cyber ethics

Internet Applications: Internet services, electronic mail(email),File Transfer, Real time user communication, Remote Login, Usenet

Word wide Web: The Web, The working Web, Web Terminology, Web Architect

UNIT 2	**Teaching Hours :12**

Hypertext Transfer Protocol(HTTP)

HTTP,HTTP version, HTTP connections ,HTTP Communication, Hypertext Transfer Protocol Secure

Hypertext Transfer Protocol State Retention :Cookies, Hypertext Transfer Protocol cache,

Evolution of Web: The Generations of Web,Web1.0,Web 2.0,Web 3.0

Big Data: A special discussion

Web IR: Information Retrieval on the web, Web Information Retrieval Tools Web Information Retrieval Architecture, Web Information Retrieval Performance Models, Google Page Rank

UNIT 3	Teaching Hours :12
Web Development Basics	
Elements of Web Element, lent side and Server side Scripting ,Model View Controller Architecture foe Web Application Development **Client-Side Technologies:**HTML Hypertext Markup Language **CSS: Cascading Style Sheets, Java Script, Bootstrap Framework, Angular JS Framework,** Server-Side Technologies; Server-side Scripting, Personal Home Pages, Node js, Server-Side Java Script	
UNIT 4	**Teaching Hours :12**
Web Application Frame Work	
Django, Ruby on Rails **Web Databases:** Web Database, Structured Query Language, Relational Data Vases, No Sql Database, Non-Relational and Distributed Data, Understanding Popular Databases Research Trends On the Web Contextual Information Retrieval, Web Mining.	
Text Books:	
1. Akshi Kumar ,"web Technology: Theory and Practice", CRC Press,2019	
Reference Text Books:	
1. Web Technology: A Developer Perspective., N P Gopalan and J Akilandeswari ,PHI ,Learning, Delhi 2. Internetworking Technologies ,An Engineering Perspective, Rahul Banerjee, PHI Learning Delhi 2011	

(21UB4C3ITP) Internet Technologies Lab

1. Demonstrate E-Mail working (Sending, Receiving, Forward).

2. How to create, organize meeting in Zoom/Googlemeet.

3. Create a form by using various attributes of the input tags (text box, multiline textbox, option button, check box).

4. Create a simple HTML page by using some of the basic tags(hyperlink, marquee, image).

5. Create a web page with multiple types of style sheet used in a single page.

6. Write a CGI sample program to send output back to the user.

7. Create a Time-Table using table tag.

8. Creation of frames in browser window using HTML.

9. Write a java script program to create dialogue boxes using alert, confirm and prompt methods.

10. Write a java script program to form Validators.

11. Write a java script program to perform four arithmetic operations: Addition, Subtraction, Multiplication and Division on two numbers.

12. Create a web site of our College.

SOFTWARE ENGINEERING

Total Teaching Hours: 48	No. Of Lecture Hours/Week:03
Max Marks:60	Credits: 03
Course Objective:	

- To provide the idea of decomposing the given problem into Analysis, Design, Implementation, Testing and Maintenance phases.
- To provide an idea of using various process models in the software industry according to given circumstances.
- To gain the knowledge of how Analysis, Design, Implementation, Testing and Maintenance processes are conducted in a software project.

Course Outcome:

CO1: To understand and apply software engineering life cycle.
CO2: Demonstrating competence in communication, planning, analysis, design, construction and deployment.
CO3: Ability to apply and work on one or more significant application domains.
CO4: Analyze and able to work as individual and as a part of multidisciplinary team to develop and deliver quality software.

UNIT 1	**Teaching Hours :12**

Introduction to Software Engineering

Evolution and impact of software engineering, software life cycle models: Waterfall, prototyping, Evolutionary and spiral models. Feasibility study, functional and Non-functional requirements, Requirements gathering, Requirements analysis and specification.

Agile Development: Agile, Agility and cost of change; Agile process, extreme programming; other agile process models.

UNIT 2	**Teaching Hours :12**

Formal Modeling and Verification

The cleanroom strategy; Functional specification; cleanroom design; cleanroom testing; formal methods: concepts; Applying mathematical notation for formal specification; formal specification languages. Software project management; The management spectrum; The management of people, product, process and project; The W5HH principle; critical practices.

Software testing strategies: A strategic approach to software testing, test strategies for conventional software, test strategies for object-oriented software, test strategies for WebApps, System testing, software testing fundamentals, white-box testing, black-box testing.

UNIT 3	Teaching Hours :12
Software Project Scheduling:	
Basic concepts and principles of project scheduling; defining task set and task network; scheduling; earned value analysis. Risk management: Reactive versus proactive strategies; software risks; risk identification; Risk projection; Risk refinement; Risk mitigation, monitoring and management; The RMMM plan, maintenance and Reengineering; software maintenance; software supportability; Reengineering; Business process reengineering; software reengineering; Reverse engineering; Restructuring; Forward engineering; The economics of reengineering.	
UNIT 4	**Teaching Hours :12**
Software Process Improvement (SPI)	
Approaches to SPI; Maturity models; The SPI process; the CMMI; The people CMM; other SPI frameworks; SPICE, Bootstrap, PSP and TSP, ISO; SPI return on investment. Software configuration Management (SCM); Basic concepts; SCM repository; The SCM process; Configuration management for web applications; SCM standards	
Text Books:	
1. Fundamentals of Software Engineering by Rajib Mall, PHI- 3rd Edition, 2009. 2. Roger S. Pressman, "Software Engineering: A Practitioner's Approach", Alternate Edition, 7th Edition, McGraw Hill, 2010.	
Reference Text Books:	
1. Software Engineering, by Ian Sommerville, Pearson Education Inc., New Delhi (2009). 2. Software Engineering: A Practitioner's Approach, by Roger S Pressman, McGraw-Hill (2005). 3. Pankaj Jalote, "An Integrated Approach to software Engineering", Narosa Publishing House PVT Ltd, Darya Ganj, New Delhi 110002	