RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Computer Science & Information Technology, V-Semester

CS IT501 - Computer Networking

Course Objectives

- To provide students with an overview of the concepts and fundamentals of computer networks
- To familiarize with the basic taxonomy and terminology of computer networking area.
- Describe how computer networks are organized with the concept of layered approach
- To experience the designing and managing of communication protocols while getting a good exposure to the TCP/IP protocol suite

Unit I

Importance of computer networks, broadcast and point to point networks, Local area networks and Wide area networks , ISO-OSI reference model, TCP/IP model , interfaces and services, Protocol data unit, connection oriented and connectionless services, service primitives, Binding Protocol Address- ARP & RARP, packet format, Encapsulation.

Unit II

Data-Link layer: - Data link layer design issues, framing , flow & error control , physical addressing, Stop & Wait protocol ,Go back N ARQ ,selective repeat ARQ ,piggybacking and pipelining ,HDLC LAN Protocol stack-Logical link control and Media Access Control sublayer, IEEE 802.2 LLC Frame format; MAC layer Protocols- static and dynamic allocation, Pure and slotted ALOHA, Carrier sense multiple access, Persistent and non persistent CSMA, IEEE standard 802.3, 802.4, 802.5, FDDI,

Unit III

The Network layer- logical addressing, classful & classless addressing, packet delivery & forwarding. unicast routing protocols , multicast routing protocols, Routing algorithm- Least Cost, Dijkstra's, Bellman-ford, Introduction to Internet protocol, IPv4 header, IPv4 Datagrams, Encapsulation, Fragmentation and Reassembly, IP routing, Subnet addressing, Subnet mask, Super netting- special case of IP addresses, Ipv6-Motivation, frame format and addressing. ICMP: Introduction, ICMP Header, ICMP message types.

Unit IV

Transport layer- TCP: Introduction ,Transport services , Process to process delivery, TCP ,congestion control algorithms, quality of service, headers, connection establishment and termination, timeout of connection establishment, maximum segment size, port no. and socket addresses, TCP timers, UDP: Introduction, UDP header, UDP checksum, UDP operations, encapsulation & decapsulation, queuing, SCTP-Services, transmission sequence number, stream identifier, stream sequence number, packet format.

Unit V

Application layer - BOOTP:-operation, packet format, DHCP:-Address allocation, configuration & packet Format, DNS: Distribution of name spaces, DNS in the internet, FTP:-Connection, Communication, command processing, TFTP, E-Mail: SMTP, POP, IMAP, SNMP. study of internetworking devices and their configuration– switches, hubs, Bridges, routers and Gateways.

References

1. ."Computer Networks" - Tanenbaum ,PHI Learning
2. "Data Communication & Networks " , Fourouzan TMH
3. "TCP/IP-Protocol suite", Forouzan, TMH 3rd edition
4. "Computer Networks and Internets", D.E.Comer, Pearson
5. "TCP/IP Illustrated" W. Richard Stevens, Volume I, Addison Wesley,
6. "Internetworking with TCP/IP Vol. I, II & III", Comer , PHI Learning.

Course Outcomes

Upon successful completion of this course the students will:

- Have agood understanding of the OSI Reference Model and its Layers
- Identify core networking and infrastructure components and the roles they serve; and given requirements and constraints, design an IT infrastructure including devices, topologies, protocols, systems software, management and security;
- Analyze the requirements for a given organizational structure and select the most appropriate networking architecture and technologies
- Specify and identify deficiencies in existing protocols, and then go onto formulate new and better protocols

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Computer Science & Information Technology, V-Semester

CSIT502 - Operating System

Course Objectives

- Learn concepts of operating systems
- Learn the mechanisms of OS to handle processes
- Study of various mechanisms involved in memory management techniques
- Gaining knowledge of deadlocks prevention and detection techniques
- Analyzing disk management functions and techniques

Unit I

Introduction to Operating Systems, Evaluation of OS, Types of operating Systems, system protection, Operating system services, Operating System structure, System Calls and System Boots, Operating System design and implementation, Spooling and Buffering.

Unit II

Basic concepts of CPU scheduling, Scheduling criteria, Scheduling algorithms, algorithm evaluation, multiple processor scheduling. Process concept, operations on processes, threads, inter process communication, precedence graphs, critical section problem, semaphores, classical problems of synchronization,

Unit III

Deadlock problem, deadlock characterization, deadlock prevention, deadlock avoidance, deadlock detection, recovery from deadlock, Methods for deadlock handling. Concepts of memory management, logical and physical address space, swapping, Fixed and Dynamic Partitions, Best-Fit, First-Fit and Worst Fit Allocation, paging, segmentation, and paging combined with segmentation.

Unit IV

Concepts of virtual memory, Cache Memory Organization, demand paging, page replacement algorithms, allocation of frames, thrashing, demand segmentation, Role of Operating System in Security, Security Breaches, System Protection, and Password Managment.

Unit V

Disk scheduling, file concepts, File manager, File organization, access methods, allocation methods, free space managements, directory systems, file protection, file organization & access mechanism, file sharing implement issue, File Management in Linux, introduction to distributed systems.

References:

1. Silberschatz ,"Operating system", Willey Pub
2. Tanenbaum " Modern Operating System" PHI Learning.
3. Dhamdhere, "System Programming and Operating System",TMH.
4. Stuart,"Operating System Principles, Design &Applications",Cengage Learning
5. Operating System : Principle and Design by Pabitra Pal Choudhury, PHI Learning

Suggested List of Experiments

1. Program to implement FCFS CPU scheduling algorithm.
2. Program to implement SJF CPU scheduling algorithm.
3. Program to implement Priority CPU Scheduling algorithm.
4. Program to implement Round Robin CPU scheduling algorithm.
5. Program to implement classical inter process communication problem(producer consumer).
6. Program to implement classical inter process communication problem(Reader Writers).
7. Program to implement classical inter process communication problem(Dining Philosophers).
8. Program to implement FIFO page replacement algorithm.
9. Program to implement LRU page replacement algorithm

Course Outcomes

Upon successful completion of this course the students will:

- Gain knowledge of history of operating systems
- Understand design issues associated with operating systems
- Gain knowledge of various process management concepts including scheduling,synchronization,deadlocks
- Understand concepts of memory management including irtual memory
- Understand issuesrelatedtofilesysteminterfaceandimplementation,diskmanagement
- Be familiar with protection and security mechanisms
- Bef amiliar with various types of operating systems including Unix

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Computer Science & Information Technology, V-Semester

Departmental Elective CSIT- 503 (A) Theory of Computation

Course Objectives

- Student learns some fundamental concepts in automata theory and designing of Finite Automata, conversion NFA to DFA. Application of Finite Automata in computer science and real world.
- Obtain minimized DFA and Application of regular expression and conversion from RE to Finite Automata and Finite Automata to Regular Expression and Proving language are not regular.
- Designing of CFG's , Construction of parse trees, finding and removing ambiguity in grammars, simplification of CFG, Conversion of grammar to Chomsky Normal Form ,Greibach normal form.
- Designing problems on Pushdown Automata and conversion of grammar to PDA, PDA to Grammar.
- Designing Turing machines, understanding the working of various types of Turing machines and study P and NP type problem.

UNIT I

Introduction of the theory of computation, Finite state automata – description of finite automata, properties of transition functions, Transition graph, designing finite automata, FSM, DFA, NFA, 2-way finite automata, equivalence of NFA and DFA, Mealy and Moore machines.

UNIT II

Regular grammars, regular expressions, regular sets, closure properties of regular grammars, Arden's theorem, Myhill-Nerode theorem, pumping lemma for regular languages, Application of pumping lemma, applications of finite automata, minimization of FSA.

UNIT III

Introduction of Context-Free Grammar - derivation trees, ambiguity, simplification of CFGs, normal forms of CFGs- Chomsky Normal Form and Greibach Normal forms, pumping lemma for CFLs, decision algorithms for CFGs, designing CFGs, Closure properties of CFL's.

UNIT IV

Introduction of PDA, formal definition, closure property of PDA, examples of PDA, Deterministic Pushdown Automata, NPDA, conversion PDA to CFG, conversion CFG to PDA.

UNIT V

Turing machines - basics and formal definition, language acceptability by TM, examples of TM, variants of TMs – multitape TM, NDTM, Universal Turing Machine, offline TMs, equivalence of single tape and multitape TMs. Recursive and recursively enumerable languages, decidable and undecidable problems – examples, halting problem, reducibility. Introduction of P, NP, NP complete, NP hard problems and Examples of these problems.

Reference Books:

1. Daniel I.A. Cohen,"Introduction to Computer Theory",Wiley India.
2. John E Hopcroft, Jeffrey D. Ullman and Rajeev Motwani, "Introduction to Automata Theory, Languages and Computation", Pearson Education.
3. K.L.P Mishra & N.Chandrasekaran,"Theory of Computer Science", PHI Learning.
4. Peter Linz, "Introduction to Automata Theory and Formal Languages", Narosa Publishing.
5. John C Martin, "Introduction to languages and the theory of computation", TATA McGraw Hill.

Course Outcomes

At the completion of the course, students will be able to...

- Convert between finite automata, regular grammars, and regular expression representations of regular languages
- Apply the pumping lemma for regular languages to determine if a language is regular
- Convert between grammars and push-down automata for context-free languages
- Determine if a language is regular or context-free
- Demonstrate that a grammar is ambiguous
- Translate a context-free grammar from one form to another
- Produce simple programs for a Turing Machine
- Explain the concept of undecidability
- List examples of undecidable problems

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Computer Science & Information Technology, V-Semester

Departmental Elective CSIT- 503 (B) Microprocessor and Interfacing

Course Objectives:

- To introduce basic concepts of microprocessor
- To introduce serial and parallel bus standards.
- To introduce programming in assembly language.
- To introduce basic concepts of interfacing memory and peripheral devices to a microprocessor.

UNIT –I:
Evolution of microprocessor, single chip micro computers, Micro processor Application, Microprocessor and its architecture, addressing modes, instruction, Instruction sets, Arithmetic and Logic Instruction, Program control instruction, Introduction –8086 family, procedure and macros, connection , Timing and Troubleshooting interrupt, 80286, 80836 and 80486 micro processor system concept.

UNIT –II:
Microprocessor Cycle, AIU, Timing and control Unit, Register data, Address bus, Pin Configuration, Intel 8086 instruction, Opcode and operands, limitation word size. Programming the microprocessor Assembly language, The Pentium and Pentium Pro Micro Processorwith features, Pentium II, Pentium III and Pentium –IV Microprocessor with software changes. Instruction set for Intel 8086, Introduction Intimation and data formats, Addressing modes, Status flags, Symbols and abbreviations, programming of microprocessors, Assembly language, high level language, areas of application of various languages, Stacks, Sub routines system, software, commands in assembly language, software Development, Debugging program, Modular programming, Structured programming, Top-down, Bottom-up design , MACRO microprogramming.

UNIT-III:
Assembly language programming with Examples like Addition of 8/16-bit Binary number, subtraction of 8/16 bit binary number, Address partitioning, addressing mode, type of addressing mode, memory and I/o interfacing, Data transfer schemes, Interfacing device and I/o devices I/o ports, Basic I/o Interfacing MDS, Micro controllers, I/o processor and co-processors ,Microcomputer Development system, Single chip micro computers, intel 8748 intel 8051, inter 8096, intel 8049intel 2920/2921, I/o processor UPI-425,UPI-41,42, Co-processor, math processor math co-processor –8087, 80287, 80387DX 803875x

UNIT –IV:

Bus Interface I/o port Addressing, decoding 8279, Programmable key board/display interface, 8254 Internal Timer, 16550 programmable communication interface A/D, 8259A Programmable Interrupt Controller, 8237 DMA Controller, Shared bus operation, disk Memory system Video display. ISA Bus, Extended ISA (EISA) and VESA Local Buses, Peripheral Component Inter Connect (Pc I) Bus, Parallel Printer interface (LPT) Universal serial Bus (USB) Accelerated graphics port (AGP),Programmable Communication interfere 8251 VSART CRT Controller 8275, 6854, Floppy disk Controller 8272, I/o processor 8089.

UNIT –V:

Memory Unit, RAM,SRAM, DRAM,ROM, PROM EPROM, EEPROM Nonvolatile RAM semiconductor Technology for memory, Shift register, Magnetic Memory, Tap, disc, main memory and secondary memory cache memory, program memory and Data Memory, Real and virtual memory Buses, memory Addressing capacity of CPU, processing speed of computer

Reference Books:

1. Douglas V Hall, “Microprocessors and interfacing –Programming & Hardware” TMH
2. Barry B. Brey, “The intel Microprocessor –8086”, Pearson Education
3. Kenneth J.Ayala,”The 8086 Microproccessor: Programming & Interfacing The PC”,CengageLearning
4. Krishna Kant,”Microprocessors and Microcontrollers”, PHI Learning
5. A.K.Ray KM Bhurchandi, “Advanced Microprocessor and peripherals” McGraw Hill
6. R.S. Gaonkar ,”Microprocessors and interfacing”, TMH

Course Outcomes:

At the completion of the course, students will be able to...

- Explain the microprocessor’sand Microcontroller’s internal architecture
- Apply knowledge and demonstrate programming proficiency using the various addressing modes and data transfer instructions of the target microprocessor and microcontroller.
- Compare accepted standards and guidelines to select appropriate Microprocessor(8085&8086) and Microcontroller to meet specified performance requirements.
- Analyze assembly language programs
- Design electrical circuitry to the MicroprocessorI/Oports in order to interface the processor to external devices.
- Evaluate assembly language programs

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Computer Science & Information Technology, V-Semester

Departmental Elective CSIT- 503 (C) Principles of Programming Languages

Course Objectives

- To introduce the major programming paradigms, and the principles and techniques involved in design and implementation of modern programming languages.
- To introduce notations to describe syntax and semantics of programming languages.
- To analyze and explain behavior of simple programs using concepts such as binding, scope, control structures, subprograms and parameter passing mechanisms.
- To introduce the concepts of concurrency control and exception handling

UNIT-I

Language Evaluation Criteria, influences on Language design, Language categories, Programming Paradigms –Imperative, Object Oriented, functional Programming, Logic Programming. Programming Language Implementation –Compilation and Virtual Machines, programming environments

UNIT-II

Data types: Introduction, primitive, character, user defined, array, associative, record, union, pointer and reference types, design and implementation uses related to these types. Names, Variable, conceptof binding, type checking, strong typing, type compatibility, named constants, variable initialization, Sequence control with Expressions, Conditional Statements, Loops, Exception handling.

UNIT-III

Subprograms and Blocks: Fundamentals of sub-programs, Scope and lifetime of variable, static and dynamic scope, Design issues of subprograms and operations, local referencing environments, parameter passing methods, overloaded sub-programs, generic sub-programs, design issues for functions overloaded operators, co routines.

UNIT-IV

Abstract Data types: Abstractions and encapsulation, introductions to data abstraction, Static and Stack-Based Storage management. heap based storage management. Garbage Collection. Object oriented programming in small talk, C++, Java, C#, PHP, Perl . Concurrency: Subprogram level concurrency, semaphores, monitors, massage passing, Java threads, C# threads

UNIT–V

Exception handling, Exceptions, exception Propagation, Exception handler in C++ and Java. Logic Programming Language : Introduction and overview of logic programming, basic elements of prolog, applicationof logic programming. Functional Programming Languages: Introduction, fundamentals. Introduction to 4GL.

Reference Books:

1. Sebesta,"Concept of Vprogramming Language", Pearson Edu.
2. Louden, "Programming Languages: Principles & Practices", Cengage Learning
3. Tucker, " Programming Languages: Principles and paradigms", Tata McGraw –Hill
4. Terrance W Pratt, "Programming Languages: Design and Implementation", Pearson Edu.
5. Cavlo Ghezzi& Mehdi Jazayeri" Programming Languages Concepts", Willey India
6 E Horowitz, "Programming Languages", 2nd Edition, Addison Wesley

Suggested List of Experiments:

1. Define a LISP function to compute sum of squares.
2. Define a LISP function to compute difference of squares.(if x > y return x2 -y2, otherwise y2 - x2).
3. Define aRecursive LISP function to solve Ackermann's Function.
4. Define a Recursive LISP function to compute factorial of a given number.
5. Define a Recursive LISP function which takes one argument as a list and returns last element of the ist. (Do not use last predicate).
6. Define a Recursive LISP function which takes one argument as a list and returns a list except last element of the list. (Do not use but last predicate).
7. Define a Recursive LISP function which takes one argument as a list and returnsreverse of the list. (Do not use reverse predicate).
8. Define a Recursive LISP function which takes two arguments first, an atom, second, a list, returns a list after.

Course Outcomes:

At the completion of the course, students will...

- Have the background for choosing appropriate programminglanguages for certain classes of programming problems
- Be able to program in an imperative (or procedural),an object-oriented, a functional, and a logical programming language
- Understand the significance of an implementation of a programming language in a compiler or interpreter
- Have the ability to learn new programming languages
- Have the capacity to express programming concepts and choose among alternative ways to express things
- Be able to design a new programming language
- Make good use of debuggers and related tools

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Computer Science & Information Technology, V-Semester

Open Elective CSIT- 504 (A) Cyber Security

UNIT 1

Introduction of Cyber Crime, Challenges of cyber crime, Classifications of Cybercrimes: E-Mail Spoofing, Spamming, Internet Time Theft, Salami attack/Salami Technique,

UNIT 2

Web jacking, Online Frauds, Software Piracy, Computer Network Intrusions, Password Sniffing, Identity Theft, cyber terrorism, Virtual Crime, Perception of cyber criminals: hackers, insurgents and extremist group etc. Web servers were hacking, session hijacking.

UNIT 3

Cyber Crime and Criminal justice: Concept of Cyber Crime and the IT Act, 2000, Hacking, Teenage Web Vandals, Cyber Fraud and Cheating, Defamation, Harassment and E-mail Abuse, Other IT Act Offences, Monetary Penalties, jurisdiction and Cyber Crimes, Nature of Criminality, Strategies to tackle Cyber Crime and Trends.

UNIT 4

The Indian Evidence Act of 1872 v. Information Technology Act, 2000: Status of Electronic Records as Evidence, Proof and Management of Electronic Records; Relevancy, Admissibility and Probative Value of E-Evidence, Proving Digital Signatures, Proof of Electronic Agreements, Proving Electronic Messages.

UNIT 5

Tools and Methods in Cybercrime: Proxy Servers and Anonymizers, Password Cracking, Key loggers and Spyware, virus and worms, Trojan Horses, Backdoors, DoS and DDoS Attacks , Buffer and Overflow, Attack on Wireless Networks, Phishing : Method of Phishing, Phishing Techniques.

Suggested Books:

1. Principles of Cyber crime, Jonathan Clough Cambridge University Press
2. John R. Vacca, Computer Forensics:Computer Crime Scene Investigation, 2nd Edition, Charles River Media, 2005
3. Cyber Law Simplified, VivekSood, Pub: TMH.
4. Cyber Security by Nina Godbole, SunitBelapure Pub: Wiley-India
5. Information Warfare: Corporate attack and defense in digital world, William Hutchinson, Mathew Warren, Elsevier.
6. Cyber Laws and IT Protection, Harish Chander, Pub:PHI.

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Computer Science & Information Technology, V-Semester

Open Elective CSIT- 504 (B) Artificial Intelligence

Course Objectives

- To present an overview of artificial intelligence (AI) principles and approaches
- Develop a basic understanding of the building blocks of AI

Unit I:

Meaning and definition of artificial intelligence, Production systems, Characteristics of production systems, Study and comparison of breadth first search and depth first search techniques, other Search Techniques like hill Climbing, Best first Search. A* algorithm, AO* algorithms etc, and various types of control strategies.

Unit II:

Knowledge Representation, Problems in representing knowledge, knowledge representation using propositional and predicate logic, comparison of propositional and predicate logic, Resolution, refutation, deduction, theorem proving, inferencing, monotonic and non-monotonic reasoning.

Unit III:

Probabilistic reasoning, Baye's theorem, semantic networks, scripts, schemas, frames, conceptual dependency, fuzzy logic, forward and backward reasoning.

Unit IV:

Game playing techniques like minimax procedure, alpha-beta cut-offs etc, planning, Study of the block world problem in robotics, Introduction to understanding, natural language processing.

Unit V:

Introduction to learning, Various techniques used in learning, Introduction to neural networks, applications of neural networks, common sense, reasoning, some example of expert systems.

References:-

1 Rich E and Knight K, “Artificial Intelligence”, TMH, New Delhi.

2 Nelsson N.J., “Principles of Artificial Intelligence”, Springer Verlag, Berlin.

Course Outcomes:

Upon successful completion of this course the students will:

- Be familiar with terminology used in this area
- Explain what constitutes "Artificial" Intelligence and how to identify systems with Artificial Intelligence.
- Know how to build simple knowledge-based systems
- Have ability to apply knowledge representation, reasoning, and machine learning techniques to real-world problems

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Computer Science & Information Technology, V-Semester

Open Elective CSIT- 504 (C) Web Technology & E Commerce

UNIT-1 Introduction to building blocks of electronic commerce: Internet and networking. Technologies, IP addressing, ARP, RARP, BOOTP, DHCP, ICMP, DNS, TFTP, TELNET.

Unit-2 Static and dynamic web pages, tiers, plug-ins, frames and forms. Exposure to Markup languages, HTML, DHTML, VRML, SGML, XML etc. CGI, Applets & Serve-lets, JSP & JAVA Beans, active X control, ASP cookies creating and reading cookies, semantic web, semantic web service ontology Comparative case study of Microsoft and JAVA technologies, web server scalability,.Distributed objects, object request brokers, component technology, Web services, Web application architectures, Browsers, Search engines.

Unit-3 Introduction to e-commerce: History of e-commerce, e-business models B2B, B2C, C2C, C2B, legal; environment of e-commerce, ethical issues, electronic data interchange, value chain and supply chain, advantages and disadvantages of e-commerce.

Unit-4 Electronic Payment Systems: Credit cards, debit cards, smart cards, e-credit accounts, e-money, Marketing on the web, marketing strategies, advertising on the web, customer service and support, introduction to m-commerce, case study: e-commerce in passenger air transport.E-Government, theoretical background of e-governance, issues in e-governance applications, evolution of e-governance, its scope and content, benefits and reasons for the introduction of e-governance

Unit-5 E-Government systems security: Challenges and approach to e-government security, security concern in e-commerce, security for server computers, communication channel security, security for client computers.

References:

1. Web Technology, Achyut Godbole, Atul Kahate, TMH

2. Henry Chan, Raymond Lee, Tharam Dillon , E-Commerce Fundamental and Applications, Willey Publication.

3. Minoli & Minoli, Web Commerce Technology Hand Book, TMH

4. Satyanarayana, E-Government, PHI

5. Uttam K: Web Technologies, Oxford University Press.

6. G. Winfield Treese, Lawrence C. Stewart, Designing Systems for Internet Commerce, Longman Pub.

7. Charles Trepper, E Commerce Strategies, Microsoft Press

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

Computer Science & Information Technology, V-Semester

CSIT505 Linux (LAB)

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

1. Understand the system calls
2. Compare between ANSI C AND C++ AND POSIX standards
3. Mapping the relationship between UNIX Kernel support for files
4. Understand Kernel support for process creation and termination and memory allocation

Overview of Unix/Linux:-
Concepts, Unix/Linux Installation Process, Hardware Requirements for Unix/Linux, Advantages of Unix/Linux, Reasons for Popularity and Success of Linux/Unix Operating System, Features of Linux/Unix Operating System, Kernel, Kernel Functions,

The Shell Basic Commands, Shell Programming:-
Shell Variables, Branching Control Structures, Loop-Control Structure, Continue and break Statements, Sleep Command, Debugging Script. Use of Linux as web-
server, file server, directory server, application server, DNS server, SMTP server, Firewall, Proxy server.

File System: -
Definition of File System, Defining Geometry, Disk Controller, Solaris File System, Disk Based File Systems, Network-Based File Systems, Virtual File systems, UFS File System, The Boot Block, The Super Block, The Inode, Tuning File System, Repairing File System.

Process Control:-
Viewing a Process, Command to display Process, Process Attributes, Process States, Process Fields, PS Commands options, PGREP, PRSTAT, CDE Process Manager, Scheduling Process, Scheduling Priorities, Changing the Priority of a time-sharing process, Killing Process.

System Security:-
Physical Security, Controlling System Access, Restricted Shells Controlling File Access, File Access Commands, Access Control List(ACLs), Setting ACL Entries, Modifying ACL entries on a file, Deleting ACL entries on a file, Restricting FTP, Securing Super User Access, Restricting Root Access, Monitoring super user Access, TCP Wrappers.

Dynamic Host Configuration Protocol: -

Introduction, DHCP Leased Time, DHCP Scopes, DHCP IP Address, Allocation Types, Planning DHCP Deployment, DHCP Configuration files, Automatic Startup of DHCP Server, Configuration of DHCP Clients, Manually Configuring the DHCP.

Case Study: -
Installation of Linux, Customization of Linux, Installation of SAMBA, APACHE,
TOMCAT, Send MAIL, Postfix, Implementation of DNS, LDAP services, Firewall, Proxy server

List of Experiments:-

1. To Study basic & User status Unix/Linux Commands.
2. Study & use of commands for performing arithmetic operations with Unix/Linux.
3. Create a file called wlcc.txt with some lines and display how many lines, words and characters are present in that file.
4. Append ten more simple lines to the wlcc.txt file created above and split the appended file into 3 parts. What will be the names of these split files? Display the contents of each of these files. How many lines will be there on the last file?
5. Given two files each of which contains names of students. Create a program to display only those names that are found on both the files.
6. Create a program to find out the inode number of any desired file.
7. Study & use of the Command for changing file permissions.
8. Write a pipeline of commands, which displays on the monitor as well as saves the information about the number of users using the system at present on a file called usere.ux.
9. Execute shell commands through vi editor.
10. Installation, Configuration & Customizations of Unix/Linux.
11. Write a shell script that accepts any number of arguments and prints them in the reverse order.
12. Write a shell script to find the smallest of three numbers that are read from the keyboard.
13. Write a shell script that reports the logging in of a specified user within one minute after he/she logs in. The script automatically terminates if the specified user does not login during a specified period of time.
14. Installation of SAMBA, APACHE, TOMCAT.
15. Implementation of DNS, LDAP services,
16. Study & installation of Firewall & Proxy server

Suggested Reading:
1. Venkatesh Murthy, “Introduction to Unix &Shell”, Pearson Edu
2. Forouzan, “Unix &Shell Programming”, Cengage Learning
3. Sumitab Das,”Unix Concept & Application”,TMH
4. Gopalan, Shivaselvan,”Beginners Guide to Unix ” PHI Learning
5. Venkateshwavle,”Linux Programming Tools Unveil`ed”, BS Publication.
6. Richard Peterson,”Linux Complete Reference”,TMH
7. Richard Peterson,”Unix Complete Reference”,TMH