

MOHANLAL SUKHADIA UNIVERSITY, UDAIPUR

FIRST YEAR SCIENCE COMPUTER SCIENCE

Paper Code	Paper & Title	Hrs/week	Max. Marks
1061	I: Digital Electronics & Introduction to Computer Architecture	2	50
1062	II: Problem Solving through C Programming	2	50
1063	III: Computer Oriented Numerical Methods	2	50
1064	IV: Practical	4	75

Note:

1. Computer Science can be offered only with Physics & Mathematics or Statistics & Mathematics.
2. Duration of theory examination will be 3hrs and practical examination will be 4 hrs
3. Each theory question paper in the annual examination shall have three sections:

Section A shall contain one compulsory question of 5 marks having 10 parts. Two parts shall be set from each unit. The candidate is required to answer each part in one or few words. **(Total: 5 Marks)**

Section B shall contain five compulsory questions of 5 marks each with internal choice. One question with internal choice will be set from each unit. The answer may be given in approximately 250 words. **(Total 25 Marks)**

Section C shall contain four descriptive questions covering all units and candidates have to answer any two questions of ten marks each. The answer may be given in approximately 500 words. There can be two parts in a question from this section. **(Total 20 Marks)**

Paper I - Digital Electronics and Introduction to Computer Architecture

Unit I

Number Systems: Binary, octal, decimal, hexadecimal and BCD number systems. Representation of positive, negative integers and real numbers. Characters digital codes: ASCII and EBCDIC coding, binary arithmetic in 1's and 2's complement.

Unit II

Boolean Algebra: Logic gates, truth table, logic expression, rules and laws of boolean algebra. Demorgan's theorems, simplification of boolean expression using Karnaugh map (upto 4 variables).

Unit III

Combinational Circuits: Adder, subtractor, comparator, decoder, encoder, multiplexer, demultiplexer. (Block diagram level only)

Flip Flops: Latches, edge-triggered flip flops, pulse triggered flip flop, R-S flip, JK master-slave flip flop, D flip flop, T flip flop.

Unit IV

Shift Registers: Shift register function, serial and parallel shift registers, bi-directional shift registers.

Counters: Asynchronous and synchronous counters, up/down counters, modulo-n counters, BCD counters.

Unit IV

Memory Organization: Basic memory cell, 1- 2-D memory, row and column address, accessing memory, different RAM and ROM types.

Overview of I/O Systems, introduction to microprocessor (8085) and microcontrollers (only organization and signals required)

Suggested Books

1. Thomas L. Floyd, Digital Fundamentals, United Book Stall New Delhi.
2. Mano M.M., Digital Logic and Computer Design, Prentice Hall of India Private Limited New Delhi.
3. Hayes J.P., Computer Organization and Architecture, Tata Mc-Graw Hill Publishing Company Limited New Delhi.
4. Mano M.M., Computer System Architecture, Prentice Hall of India Private Limited New Delhi.

Paper II – Problem Solving through C programming

Unit I

Overview of computer system architecture: Simple model of a computer to explain how it executes algorithms, CPU, memory, I/O units main and secondary memory. CPU structure, machine instruction, operation code and operand location, instruction counter, fetch and execute cycles in a computer. Arithmetic and logic operations.

Memory locations, address and contents, read and write operations. Difference between low level and high level languages, hierarchy of programming languages.

Computer generations: First, second, third, fourth, fifth generations.

Types of computers: Mainframe, Mini, super computers, micro computers.

Unit II

Algorithm and algorithm development: Definition and properties of algorithm, flow chart symbols, conversion of flow chart to language, example of simple algorithms. Program design, errors : syntax error, runtime error, logic error, debugging, program verification, testing, documentation and maintenance.

Unit III

Introduction to C: Variables and arithmetic expressions, the for statement, symbolic constants, character input and output, arrays, functions, arguments- call by value, character arrays, external variables and scope.

Types, Operators and Expressions: Variable names, data type and sizes, constants, declarations, arithmetic operators, relational and logical operators, type conversions, increment and decrement operators, bitwise operators, assignment operators and expressions, conditional expressions, precedence and order of evaluation.

Control Flow: Statements and blocks, if-else, else-if, switch, loops- while and for, loops- do-while, break and continue, goto and labels.

Unit IV

Functions and Program Structure: Basics of function, functions returning non-integers, external variables, scope rules, header files, static variables, register variables, block structure, initialization, recursion, the C preprocessor.

Pointer and Arrays: Pointers and addresses, pointers and function arguments, pointers and arrays, address arithmetic. Character pointers and functions, pointer arrays: pointers to pointers, multi-dimensional arrays, pointers vs. multi-dimensional arrays. Pointers to functions. Complicated declarations.

Unit V

Structures: Basics of structures, structures and functions, arrays of structures, pointers to structures, self-referential structures, table lookup, typedef, unions, bit-fields.

Input and Output: Standard input and output. Formatted output- printf, variable length argument lists. Formatted input- scanf, file access, error handling- stderr and exit, line input and output, miscellaneous functions.

Suggested Books

1. Deendayalu R., Computer science Volume I and II, Second Edition, Tata McGraw-Hill Publishing Company Limited, New Delhi.

2. Rajaraman V., Fundamentals of computers, Second Edition, Prentice Hall of India Private Limited, New Delhi.
3. Kernighan B.W. and Ritchie D.M., The C Programming Language, Prentice Hall of India Private Limited New Delhi.
4. Drogmey R., How to solve it by computers. Prentice Hall of India Private Limited , New Delhi.

Paper III -Computer Oriented Numerical Methods

Unit I

Computer Arithmetic : Floating point representation of numbers, arithmetic operations with normalized floating point numbers and their consequences ,error in number representation, pitfalls in computing. Iterative Methods : successive approximation, bisection method, false position, Newton Raphson method, comparison of iterative methods, Baristow's method for finding complex roots.

Unit II

Solution of simultaneous linear equations : Gauss elimination method, pivoting ill conditioned equations, refinement of solution, Gauss Seidel iterative method, comparison of direct and iterative methods.

Unit III

Solution of ordinary differential equations : Euler's method, Taylor series method, local and global error analysis, Runge-Kutta methods, predictor-corrector methods, stability of solution.

Unit IV

Interpolation and approximation : Langrange's interpolation, Polynomial interpolation, difference table, truncation error in interpolation, linear regression, polynomial fitting and other curve fitting techniques, approximation of function by Taylor series and Chebyshev series.

Unit V

Numerical differentiation and integration: Differentiation formulae based on polynomial fits, pitfalls in differentiation. Integration : trapezoidal, Simpson rules, Guassian quardature.

Suggested Books

1. Rajaraman V., Computer Oriented Numerical Methods, Peentice Hall of India Private Limited.
2. Krishnamurthy , Computer based numerical algorithms, East West Press.

Paper IV Practical

Note: Students are required to perform all the experiments. In the examination three exercises selecting one from each part will be set.

Marks Distribution:

Part A: 12 Marks

Theory: 6 Marks & Logic diagram and verification on digital kits: 6

Part B: 12 Marks

Algorithm and flowchart: 3 Marks, Program coding: 3 Marks

Program execution, result and documentation: 6

Part C: 24 Marks

Algorithm and flowchart: 6 Marks, Program coding: 6 Marks

Program execution, result and documentation: 12 Marks

Viva: 12 Marks,

Practical Record: 15 Marks (Regular submission of practical records and evaluation by teachers, Regular attendance in the practical classes)

Part A: Digital Electronics

1. To study the function of Basic logic gates and verify their truth table AND, OR, NOT, NAND, NOR, XOR, XNOR.
2. To study the application of AND, OR, NAND, XOR gates for getting digital signals
3. (a) The study of different logical expression and their simplifications.
(b) To familiarize and verify the Boolean algebraic theorems.
4. To study the different arithmetic circuits.
(a) Half adder and subtractor

- (b) Full adder and subtractor
- 5. To study the BCD to binary and Binary to BCD code converter.
- 6. To study the binary- to -gray and Gray-to binary code converter.
- 7. Study of the encoder - circuits.
 - (a) Decimal to BCD- encoder
 - (b) Octal to binary encoder.
- 8. Study of decoder circuits.
 - (a) BCD - to - decimal decoder.
 - (b) BCD - to - 7 Segment decoder.
- 9. To study the Flip flop circuits using gates.
 - (a) R-S flip flop
 - (b) J-K flip flop
 - (c) Master slave J-K flip flop
 - (d) D- flip flop
- 10. To study the R-S, J-K and D-flip flop ICs
- 11. To study the shift registers and ripple counter.
- 12. To study the asynchronous counter using flip flop ICs.
- 13. To study asynchronous counter ICs.
- 14. To study synchronous counter ICs.
- 15. To study synchronous counter using flip flop ICs.

Part B : Programming in C

- 1. Addition and subtraction of 8 bit numbers
- 2. Find 1's and 2's complement of 8 and 16 bit numbers
- 3. Shifting left/right of 8/16 bit numbers
- 4. Logical operation such as - masking and setting of specific bits
- 5. To find larger/smaller of two numbers
- 6. Sum of series of 8 bit numbers
- 7. To arrange a series of numbers in ascending/descending order
- 8. To multiply, divide two 8 bit numbers

9. Multi byte addition, subtraction

Part C : Numerical Methods

Numerical methods implementation using any programming language.

1. Program for addition, subtraction, multiplication of matrices.
2. Program to find transpose, inverse of matrix.
3. Program to test symmetry of matrix.
4. Program to find root of an equation by
 - (a) Bisection method
 - (b) False position method
 - (c) Newton Raphson method.
5. Solution of differential equation by
 - (a) Runge Kutta method
 - (b) Predictor corrector method
6. To solve simultaneous equations by
 - (a) Gauss Siedel method
 - (b) Gauss elimination method
7. To find integral of a function using
 - (a) Trapezoidal method
 - (b) Simpson method
8. To interpolate a function using Langrange's interpolation method.
9. To find complex roots of an equation using Bairstow's method.
10. To fit linear and exponential curves using least square fit method.

SECOND YEAR SCIENCE 2007-08

COMPUTER SCIENCE

Paper Code	Paper & Title	Hrs/week	Max. Marks
2061	I: Data Structure Using C	2	50
2062	II: Database Management System	2	50
2063	III: Computer Organization	2	50
2064	IV: Practical	4	75

Note:

- 1 Computer Science can be offered only with Physics & Mathematics or Statistics & Mathematics.
- 2 Duration of theory examination will be 3hrs and practical examination will be 4 hrs
- 3 Each theory question paper in the annual examination shall have three sections:

Section A shall contain one compulsory question of 5 marks having 10 parts. Two parts shall be set from each unit. The candidate is required to answer each part in one or few words. **(Total: 5 Marks)**

Section B shall contain five compulsory questions of 5 marks each with internal choice. One question with internal choice will be set from each unit. The answer may be given in approximately 250 words. **(Total 25 Marks)**

Section C shall contain four descriptive questions covering all units and candidates have to answer any two questions of ten marks each. The answer may be given in approximately 500 words. There can be two parts in a question from this section. **(Total 20 Marks)**

Paper I - Data Structure Using C

Unit I

Linear Structure: Arrays, records, stack, operation on stack, implementation of stack as an array, queue, operations on queue, implementation of queue.

Unit II

Linked Structure : List representation, operations on linked list - get node and free node operation, implementing the list operation, inserting into an ordered linked list, deleting, circular linked list, doubly linked list.

Unit III

Tree Structure : Binary search tree, inserting, deleting and searching into binary search tree, implementing the insert, search and delete algorithms, tree traversals

Unit IV

Graph Structure : Graph representation - Adjacency matrix, adjacency list, adjacency multilist representation. Orthogonal representation of graph . Graph traversals - bfs and dfs. Shortest path, all pairs of shortest paths, transitive closure, reflexive transitive closure.

Unit V

Searching and sorting : Searching - sequential searching, binary searching, hashing. Sorting - selection sort, bubble sort, quick sort, heap sort, merge sort, and insertion sort, efficiency considerations.

Suggested Book

1. Horowitz E Sartaj Sahni, Fundamentals of data structure, Galgotia Publication Private Limited., New Delhi.

Paper-II: Database Management Systems

UNIT-I

Introduction : Purpose of the data base system, data abstraction, data model, data independence, data definition language, data manipulation language, data base administrator, data base users, overall structure.

ER Model : entities, mapping constrains, keys, E-R diagram, reduction E-R diagrams to tables, generatio, aggregation, design of an E-R database scheme.

UNIT-II

Relational Model : The catalog, base tables and views. Relational Data Objects - Domains and Relations: Domains, relations, kinds of relations, relations and predicates, relational databases.

Relational Data Integrity - Candidate keys and related matters: Candidate keys. Primary and alternate keys. Foreign keys, foreign key rules, nulls. Candidate keys and nulls, foreign key and nulls.

UNIT-III

The SQL Language: Data definition, retrieval and update operations. Table expressions, conditional expressions, embedded SQL.

Views: Introduction, what are views for, data definition, data manipulation, SQL support.

UNIT-IV

Network model : basic concepts, data structure diagrams, DBTG CODASYL model, DBTG data retrival facility, DBTG update facility, DBTG set processing facility, mapping networks to file, networks system.

Hierarchical model : basic concepts, tree structure diagrams, data retrieval facility, update facility, virtual records, maping hierarchical to files, hierarchical system.

UNIT-IV

File and system structure : overall system structure, file organisation, logical and physical file organization, sequential and random, hierarchical, inverted, multi list, indexing and hashing, B-tree index files.

Suggested Book

1. Date C.J., Database Systems, Addison Wesley.
2. Korth, Database Systems Concepts, McGraw Hill.

Paper III- Computer Organization

Unit I

Basic Computer Organization: Instruction codes, direct and indirect address, timing and control signal generation, instruction cycle, memory reference instructions, input output instructions.

Register Transfer and Micro Operations: Bus and memory transfers, three state bus buffers, binary adder, binary incremter, arithmetic circuit, logic and shift micro operations, ALU.

Unit II

Central Processing Unit: General register organization, memory stack, one address, two address instructions, data transfer, arithmetic, logical and shift instructions, software and hardware interrupts (only brief introduction), arithmetic and instruction pipelines.

Unit III

Computer Arithmetic: Addition and subtraction with signed magnitude data, multiplication algorithms, hardware algorithm and booth algorithm, division algorithm.

Input Output Organization: Asynchronous data transfer- handshaking, asynchronous serial transfer, interrupt initiated I/O, DMA transfer, interfacing, peripherals with CPU (introduction), keyboard, mouse, printer, scanner, network card.

Unit IV

Memory Organization: ROM, RAM, hard disk, CD-ROM, Cache memory- direct mapping scheme, virtual memory concept.

Unit V

Assembly language Programming: 8085 assembly instructions and assembly language programming.

Suggested Book

1. Mano M., Computer System Architecture, Pearson Education.

Paper IV: Practical

Note : Students are required to perform all the experiments. In the examination two exercises selecting one from each part will be set.

Marks Distribution:

Part A: 24 Marks

Algorithm and flowchart: 8 Marks, Program coding: 8 Marks, Program execution, result and documentation: 8 Marks

Part B: 12 Marks

Program coding: 6 Marks, Program execution, result and documentation: 6 Marks

Part-C: 12 marks

Program coding : 6 Marks, Program execution, Result and Documentation: 6

Viva: 12 Marks

Record: 15 Marks

Part A: Data structure programming using C.

1. Programs using array-one dimensional , two dimensional.
2. Programs on matrix addition, subtraction, multiplication, transpose.
3. Programs on records, variant records, array of records.
4. Program to create a linked list using pointers write procedure to: insert cell in middle of list delete cell from middle of list printing elements of list.
5. Program for creations /maintaining binary search tree .Procedures to perform following operations on binary search tree.
 - (a) search an element
 - (b) insert an element
 - (c) delete an element
 - (d) print elements in pre order, post order, inorder form.
6. Programs for performing basic operations on stacks and queues .
7. Programs using procedure and functions (recursive and non recursive)
8. Programs on sorting array using
 - (a) selection sort
 - (b) insertion sort
 - (c) bubble sort
 - (d) quick sort
 - (e) heap sort
9. Program for searching an element in array using linear and binary search.
10. Simple file processing program on sequential file of records

Part-B

Database Management

1. Making queries using of SQL commands
2. Simple payroll Program
3. Simple Library management
4. Simple Inventory Control program
5. Result preparation program
6. Bus/Tour Ticketing program

PART-C: Assembly Language Programming for 8085 Microprocessor

1. Addition and subtraction of 8 bit numbers
2. Find 1's and 2's complement of 8 and 16 bit numbers
3. Shifting left/right of 8/16 bit numbers
4. Logical operation such as - masking and setting of specific bits
5. To find larger/smaller of two numbers
6. Sum of series of 8 bit numbers
7. To arrange a series of numbers in ascending/descending order
8. To multiply, divide two 8 bit numbers

THIRD YEAR SCIENCE COMPUTER SCIENCE

Paper Code	Paper & Title	Hrs/week	Max. Marks
3061	I: Computer Networks	2	50
3062	II: Object Oriented Programming using C++	2	50
3063	III: Computer Operating System	2	50
3064	IV: Practical	4	75

Note:

- 1 Computer Science can be offered only with Physics & Mathematics or Statistics & Mathematics.
- 2 Duration of theory examination will be 3hrs and practical examination will be 4 hrs
- 3 Each theory question paper in the annual examination shall have three sections:

Section A shall contain one compulsory question of 5 marks having 10 parts. Two parts shall be set from each unit. The candidate is required to answer each part in one or few words. **(Total: 5 Marks)**

Section B shall contain five compulsory questions of 5 marks each with internal choice. One question with internal choice will be set from each unit. The answer may be given in approximately 250 words. **(Total 25 Marks)**

Section C shall contain four descriptive questions covering all units and candidates have to answer any two questions of ten marks each. The answer may be given in approximately 500 words. There can be two parts in a question from this section. **(Total 20 Marks)**

PAPER-I: Computer Networks

Unit I

Introduction: Use of computer networks, network hardware, network software, reference models, example networks, example data communication services, network standardization.

Unit II

The Physical Layer: The theoretical basis for data communication, transmission

media, wireless transmission, the telephone system, narrowband ISDN, broadband ISDN and ATM, cellular radio, communication satellites.

Unit III

The Data Link Layer: Data link layer design issues, error detection and correction, elementary data link protocols, sliding window protocols, protocol specification and verification, example data link protocols.

Unit IV

Network Layer: Network layer design issues, routing algorithms, congestion control algorithms, internetworking, the network layer in the internet, the network layer in the ATM networks.

Unit V

The Transport Layer: The transport service, elements of transport protocols, a simple transport protocol, the internet transport protocol (TCP and UDP), the ATM AAL layer protocols, performance issues.

Suggested Book

1. Tanenbaum A.S., Computer Networks, Prentice Hall of India Private Limited New Delhi.

PAPER-II: Object Oriented Programming using C++

Unit I

Principles of Object Oriented Programming (OOP): Object oriented programming paradigm, basic concepts of object oriented programming, benefits of OOP.

Beginning with C++: What is C++? Applications of C++. A simple C++ program. More C++ statements. An example with class, structure of C++ program.

Unit II

Expressions and Control Structures: Operators in C++, scope resolution operator, member dereferencing operators, memory management operators, manipulators, type cast operator. Expressions and implicit conversions, operator overloading, operator precedence, control structures.

Functions in C++: The main function, function prototyping, call by reference, return by reference, inline functions, default arguments, const argument, function overloading, friend and virtual functions.

Unit III

Classes and Objects: Specifying a class, defining member functions. A C++ program with class, making an outside function inline, nesting of member functions, private member functions, arrays within a class, memory allocation for objects. Static data members, static member functions. Arrays of objects, objects as a function argument, friendly functions, returning objects, const member functions, pointers to members.

Unit IV

Constructors and Destructors: Constructors, parameterized constructors, multiple constructors in a class, constructors with default arguments, dynamic initialization of objects, copy constructor, dynamic constructors, constructing two-dimensional arrays, destructors.

Inheritance: Extending Classes: Defining derived classes, single inheritance, making a private member, inheritable, multi level inheritance, multiple inheritance, hierarchical inheritance, and hybrid inheritance. Virtual base classes, abstract classes, constructors in derived classes, member classes, nesting of classes.

Unit V

Pointers, Virtual Functions and Polymorphism: Pointers to objects, pointers to derived classes, virtual functions, pure virtual functions.

Suggested Book

1. Balaguruswamy E., Object Oriented Programming with C++, Tata Mc-Graw Hill New Delhi.

Paper III – Operating Systems

Unit I

Introduction: What is an operating system? Mainframe, desktop, multiprocessor, distributed, clustered, real-time and handheld systems.

Operating System Structures: System components, operating system services, system calls, systems programs, system structure, virtual machines.

Unit II

Process: Process concept, process scheduling, operations on processes, cooperating processes. Inter process communication.

CPU Scheduling: Basic concepts, scheduling criteria, scheduling algorithms, algorithm evaluation.

Unit III

Process Synchronization: The critical section problem, semaphores, classical

problems of synchronization.

Deadlocks: Deadlock characterization, methods for handling deadlocks. Deadlock prevention, avoidance and detection. Recovery from deadlocks.

Unit IV

Memory Management: Swapping, contiguous memory allocation, paging, segmentation, segmentation with paging.

Virtual Memory: Demand paging, page replacement, allocation of frames, thrashing.

Unit V

Linux: History, design principles, kernel modules, process management, scheduling, memory management, file systems, input and output, inter process communication, network structure, security.

Suggested Book

1. Silberschatz G.G., Operating System Concepts, John Wiley & Sons Inc.

Paper IV: Practical

Note: Students are required to perform all the experiments. In the examination two exercises selecting one from each part will be set.

Marks Distribution:

Part A: 24

Program coding: 12

Program execution, result and documentation: 12

Part B: 24

Algorithm and flow chart: 8

Program coding: 8

Program execution, result and documentation: 8

Viva: 12

Record: 15

Part A

Object Oriented Programming (using C++) Lab

- Write a function `power()` to raise a number `m` to a power `n`. The function takes a double value for `m` and int value for `n`, and returns the result correctly. Use a default value of 2 for `n` to make the function to calculate squares when this argument is omitted. Write a main that gets the value of `m` and `n` from user to test the function.
 - Write a function that performs the same operation as above but takes an int value for `m`. Both the functions should have the same name. Write a main that calls both the functions. Use concept of overloading.
- Define a class to represent a bank account. Include the following members. Data members: Name of the depositor, account number, type of account, balance amount in the account. Member functions: To assign initial values, to deposit an amount, to withdraw an amount after checking the balance. Write a main program to test the program.
- Write a class to represent a vector (a series of float values). Include member functions to perform the following tasks: To create the vector, to modify the value of a given element, to multiply by a scalar value, to display the vector in the form (10, 20, 30, ...). Write a program to test your class.
- A bookshop maintains the inventory of books that are being sold at the shop. The list includes details such as author, title, price, publisher and stock position. Whenever a customer wants a book, the sales person inputs the title and author and the system searches the list and displays whether it is available or not. If it is not, an appropriate message is displayed. If it is, then system displays the book details and requests for the number of copies required. If the requested copies are available, the total cost of the requested copies is displayed, otherwise the message "Required copies not in stock" is displayed. Design a system using a class called books with suitable member functions and constructors. Use new operator in constructors to allocate memory space required.
- Create a class `FLOAT` that contains one float data member. Overload all the four arithmetic operators so that they operate on the objects of `FLOAT`.
- Create a class `matrix` and write member functions to implement various matrix operations.
- Create a class `complex` for complex numbers and write member functions to implement various operations between complex numbers.
- Create a class `rational` for rational numbers and write member functions to implement various operations between rational numbers.
- Create a class `stack` and write member functions to implement various stack operations.
- Create a class `queue` and write member functions to implement various queue operations.
- Create a class `list` and write member functions to implement various list operations.
- Assume that a bank maintains two kinds of accounts for customers, one called as savings account and the other as current account. The savings account provides compound interest and withdrawal facilities but no

cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed.

Create a class account that stores customer name, account number and type of account. From this derive the classes cur_acct and sav_acct to make them more specific to their requirements. Include necessary member functions in order to achieve the following tasks: (a) Accept deposit from a customer and update the balance, (b) display the balance, (c) compute and deposit interest, (d) permit withdrawal and update the balance and (e) check the minimum balance, impose penalty, if necessary and update the balance. Do not use any constructors. Use member functions to initialize the class members.

13. Create a base class called shape. Use this class to store two double type values that could be used to compute the area of figures. Derive two specific classes triangle and rectangle from the base shape. Add to the base class, a member function get_data() to initialize base class data members and another member function display_area() to compute and display the area of figures. Make display_area() as a virtual function and redefine this function in the derived classes to suit their requirements.

Using these three classes, design a program that will accept dimensions of a triangle or rectangle interactively and display the area.

Part B:Linux and Network Lab

Write shell scripts for the following.

1. Accept a pattern and a filename. Check whether the pattern is existing in the file. If found, display “The pattern <pattern> exists in the file <filename>”. If it is not found, display the error message as “The pattern <pattern> is not found in the file <filename>”.
2. To display the following information.
 - (a) Your login directory is <login directory name>,
 - (b) You will get the primary prompt as <primary prompt>,
 - (c) Now the date and time is <date> and
 - (d) You can execute files in the directories <path>.
3. Accept the name of a file, subdirectory name (which this shell script has to create) and the directory under which this specified subdirectory has to be created. Create the new subdirectory at the directory to this newly created directory.
4. Accept the following information and storing it in a file: Cassette No., movie name, language, price and date of release. The user should be given facility to enter as many records as he wants.
5. The first parameter has to be a directory name. It should be followed by filenames. There can be more than 10 filenames. These specified files

should be copied to the directory name passed as the first parameter. The script should also check whether the directory is already existing. If not, it has to be created. The program should also check for the existence of the individual files, whether they are existing and readable.

6. To delete the specified record from the library file. It should accept a book number, check whether it is existing. If so, display the current details, ask for confirmation and then delete it from the file.
7. Accept a word and display its length.
8. Accepting a word from the user and display the reverse of the word.
9. Accept a filename. Assign a read and write permission to others for that file, only if it is not having these permissions at present.
10. Temperature conversion from centigrade to Fahrenheit and vice versa.
11. Accept a number n and a word. It then prints the word n times, one word per line or just on one line (the option should be available).
12. To print the usual date output as default and has options for printing just the time, just the day-month-year or just the day of the week.
13. To find out whether the given file is a block, character or executable file.
14. Accepts a login name and reports when person logs in. It also sends a greeting to the person.
15. To change your prompt to your login name.

Networking Lab

1. Configure peer to peer Network under Linux
2. Configure Switched Ethernet LAN
3. Configure DHCP
4. Configure File Server