ORDINANCE No. 149

MASTER OF SCIENCE

(COMPUTER SCIENCE)

1. The degree of Master of Science in Computer Science shall be of Two academic years.

2. The eligibility for admission shall be as follows;

   "A candidate who has passed B.Sc. with mathematics as one of the subject with two of the subjects from among: Physics, Chemistry, Information Technology, Electronics, Computer Science, Electronics Equipment maintenance and the subject permitted by the Board of Studies (Computer Science and Information Technology) & faculty / Three years B.C.A. degree or B.Sc. Industrial Electronics or Applied Electronics."

3. The examination shall be comprising of Theory Examination, Practical Examination and Sessionals, as per the "Scheme of Examination " recommended by the Board of Studies & Faculty.

4. Detail of subjects to be taught in two-year curriculum pattern and examination scheme for each year course shall be formed and implemented as per recommendation of Board of Studies & Faculty.

5. Requirement for examination and Admission: i) Examination will be conducted by Pt. Ravishankar Shukla University, Raipur ii) Examination shall be in Theory and Practical as stipulated. iii) A candidate will be permitted to appear for the examination only if

   A) The candidate has put up a minimum attendance of 75% of the lectures on each of the subjects as well as in practical classes provided and communicated, University may condone shortfall as required by rules.

   B) He/She obtains a certificate from the Head of institutes of having satisfactorily completed the course of study prescribed in the subject his/her conduct has been satisfactory, and his/her performance in its internal tests has been satisfactory.

6. The provisions of this course in respect of examination, attendance, results and grades are subject to alteration from time to time and shall also confirm to guidelines of AICTE/UGC/MHRD/Govt. of India / Statutes and Ordinance of this University. Provisions for Sessional rules etc., if applicable for a given examination and not mentioned otherwise, will be as per the existing provisions in the Pt.R.S.S. University Ordinance No.85.

7.   (a)   There shall be no classification of the examinees successful in each examination of the course.

   (b)   The classification of examinees after having passed all the examinations as per the "Scheme of the examination" shall be made as follows:-

      (i)   The examinees who have obtained 75% or more marks in the aggregate considering all the examination taken together shall be placed in First Division with Honours.

      (ii)   The examinees who have obtained 60% or more marks but less than 75% in the aggregate considering all the examination taken together shall be placed in the First Division.

      (iii)   The examinees who have obtained less than 60% marks and more than 48% in the aggregate considering all the examination taken together shall be placed in the Second Division.

      (iv)   An examinee must secure minimum 48% of the total aggregate to be declared successful in any examination, otherwise he/she will be declared "FAIL"

8.   As soon as possible, after the examination the Executive Council shall publish a list of successful examinees arranged in the three divisions, the names of the examinees who obtained the First ten places in First Division with Honours / First Division being arranged in Order of Merit. Any such provisions, applicable but not mentioned here will be treated as per the existing provisions/rules of the Revised Ordinance No. 22 of Pt. Ravishankar Shukla University, Raipur.

9.   **SCRUTINY OF MARKS & REVALUATION:**

A candidate, whose result has been declared, may apply to the Registrar for the scrutiny of his/her marks in the prescribed form and the rechecking of his result, or for revaluation of his answer books within thirty days of declaration of the result, according to the provisions laid down in Ordinance No. 5 and 6. The fee for scrutiny of marks and for revaluation shall be as per University rules.

10.   Medium of Instructions and Examination will be English.

SCHEME OF TEACHING AND EXAMINATION 2004-2005

MASTER OF SCIENCE

(COMPUTER SCIENCE)

FIRST YEAR (PREVIOUS)

Subject Code	Subjects	Teaching Load per week			Examination Marks					
					Max. Marks			Min. Marks		
		L	T	P	Th	Pr	Ses	Th	Pr	Ses
Paper - I	Discrete Mathematics & Scientific Computing	3	2	-	100	-	50	40	-	30
Paper - II	Advanced Operating System	3	2	-	100	-	50	40	-	30
Paper - III	Computer Architecture &System Programming	3	2	-	100	-	50	40	-	30
Paper - IV	Computer Network & Internet Technology	3	2	-	100	-	50	40	-	30
Paper - V	Data Structure and Design Analysis of Computer Algorithm with C & C++	3	2	-	100	-	50	40	-	30
	Practical I	-	-	5x2	-	100	-	-	50	-
	Practical II	-	-	5x2	-	100	-	-	50	-
	TOTAL	30	20	20	500	200	250	200	100	150

The candidates not obtaining the desired minimum passing marks and matters related to it will be treated as per the provision of Revised Ordinance No.22

SCHEME OF TEACHING AND EXAMINATION 2004-2005

MASTER OF SCIENCE

(COMPUTER SCIENCE)

SECOND YEAR (FINAL)

Subject Code	Subjects	Teaching Load per week			Examination Marks					
					Max. Marks			Min. Marks		
		L	T	P	Th	Pr	Ses	Th	Pr	Ses
Paper - VI	Computer Design and Automata Theory	3	2	-	100	-	50	40	-	30
Paper - VII	Software Engineering and Software Project Management	3	2	-	100	-	50	40	-	30
Paper - VIII	Artificial Intelligence and Neural Networks	3	2	-	100	-	50	40	-	30
Paper - IX	Data Base Management System	3	2	-	100	-	50	40	-	30
Paper - X	Computer Graphics and Multimedia System	3	2	-	100	-	50	40	-	30
	Practical III	-	-	5x2	-	100	-	-	50	-
	Project	-	-	5x2	-	100	-	-	50	-
	TOTAL	30	20	20	500	200	250	200	100	150

The candidates not obtaining the desired minimum passing marks and matters related to it will be treated as per the provision of Revised Ordinance No.22

HEMCHAND YADAV VISHWAVIDYALAYA, DURG (C.G.)

Website -www.durguniversity.ac.in, Email - durguniversity@gmail.com

SCHEME OF EXAMINATION
&
SYLLABUS
of
M.Sc. (Computer Science)

Semester Examination
For

Session 2024-2025

(Approved by Board of Studies)
Effective from June 2024

SCHEME OF TEACHING AND EXAMINATIONS
MASTER OF SCIENCE IN COMPUTER SCIENCE

Program Outcome

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

- The Masters programme aims to impart a sound understanding of the advanced principles of Computer Science.
- It provides sufficient depth and breadth of experience in up-to-date methodologies.
- An exhaustive treatment of selected research-based topics, to significantly advance a student's career prospects within the IT industry, and/or equip the student to undertake research in Computer Science.
- The programme provides theory, elective, practical, research paper, Industrial Plant Training and software project courses as a core courses.

FIRST SEMESTER

Subject Code	SUBJECTS	Teaching Load Per Week			Credit L+ (T+P)/2	Examination Marks							
						Max. Marks				Min. Marks			
		L	T	P		Th	Ses	Pr	Total	Th	Ses	Pr	Tot
Paper 1	Mathematical Foundation of Computer Science	3	2	-	4	100	50	-	150	40	30		70
Paper II	Advance Operating System	3	2	-	4	100	50	-	150	40	30		70
Paper III	Data Structure through algorithms using 'C'	3	2	-	4	100	50	-	150	40	30		70
Paper IV	Object Oriented Programming using 'C++'	3	2	-	4	100	50	-	150	40	30		70
Paper V	Computer System Architecture	3	2	-	4	100	50	-	150	40	30		70
Practical I	Programming Lab Based on Paper III			3x2	3		25	100	125		15	50	65
Practical II	Programming Lab Based on Paper- IV	-	-	3x2	3	-	25	100	125	-	15	50	65
TOTAL		15	10	12	26	500	300	200	**1000**	200	180	100	480

Paper I : Mathematical Foundation of Computer Science

Max Marks:100   **Min Marks:40**

NOTE: - The Question Paper setter is advised to prepare unit-wise question with the provision of internal choice.

Course Outcome:
Student will be able to-

- Understand the concepts of Digital Electronics.
- Apply the concept of Automata Theory
- Solve the problems with Optimization Methods
- Use the hypothetical testing
- Familiar with the graph theory and its applications

UNIT – I: Mathematical Logic, Sets Relations and functions
Mathematical Logic : Notations, Algebra of Propositions & Propositional functions, logical connectives, Truth values & Truth table Tautologies & Contradictions, Normal Forms, Predicate Calculus, Quantifiers.
Set Theory: Sets, Subsets, Power sets, Complement, Union and Intersection, De-Morgan's law Cardinality, relations: Cartesian Products, relational Matrices, properties of relations equivalence relation functions: Injection, Surjection, Bijection, Composition, of Functions, Permutations, Cardinality, the characteristic functions recursive definitions, finite induction.

UNIT – II: Lattices & Boolean Algebra
Lattices: Lattices as Algebraic System, Sub lattices, some special Lattices (Complement, Distributive, Modular).
Boolean Algebra: Axiomatic definitions of Boolean algebra as algebraic structures with two operations, Switching Circuits.

UNIT – III: Groups Fields & Ring
Groups: Groups, axioms, permutation groups, subgroups, co-sets, normal subgroups, free subgroups, grammars, language).
Fields &Rings: Definition, Structure, Minimal Polynomials, Irreducible Polynomials, Polynomial roots & its Applications.

UNIT - IV: Graphs
Graphs: Simple Graph, Multigraph &Psuedograph, Degree of a Vertex, Types of Graphs, Sub Graphs and Isomorphic Graphs, Operations of Graphs, Path, Cycles and Connectivity, Euler and Hamilton Graph, Shortest Path Problems BFS(Breadth First Search, Dijkastra's Algorithm, Representation of Graphs, Planar Graphs, Applications of Graph Theory.

UNIT - V:Trees
Trees: Trees, Properties of trees, pendant vertices in a tree, center of tree, Spanning tree, Binary tree, Tree Traversal, Applications of trees in computer science.

BOOKS RECOMMENDED:
1. A text book of Discrete Mathematics – By Swapan Kumar Sarkar.(S.Chand& company Ltd.).
2. Discrete Mathematical structure with - By J.P Trembly & R.P. Manohar.
   Applications to computerscience
3. DiscreteMathematics   -By K.A Ross and C.R.Bwritht.
4. DiscreteMathematicsStructures   -By Bernard Kohman& RobertC.Bushy.
   for computerscience
5. DiscreteMathematics   -By Seymour Lipschutz Mare Lipson. Tata McGraw-Hill Edition.

Suggested Digital Platforms Web Links:
https://onlinecourses.nptel.ac.in/noc22_cs123/preview
https://onlinecourses.nptel.ac.in/noc22_cs85/preview

Paper II: Advance Operating Systems

Max Marks:100   **Min Marks :40**

NOTE :- The Question Paper setter is advised to prepare unit-wise question with the provision of internal choice.

Course Outcome:

Student will be able to-

- Design and understand the following OS components: System calls, Schedulers, Memory management systems, Virtual Memory and Paging systems.
- Evaluate, and compare OS components through instrumentation for performance analysis.
- Analyze the various device and resource management techniques for timesharing and distributed systems.
- Develop and analyze simple concurrent programs using transactional memory and message passing, and to understand the trade-offs and implementation decision.

UNIT-I
Introduction

What is operating system, basic concept, terminology, batch processing, spooling, multiprogramming, time sharing, real time systems, protection, multiprocessor system, operating system as resource manager, process view point, memory management, process management, device management and information management, other views of operating system, historical, functional job control language and supervisor service control.

UNIT-II
Advanced Processor Management Features

Multi- threaded operating system architecture micro-kernels operating system architecture multiple operating system- subsystem and environments, client-server architecture, protected mode software architecture ,visual machine- JAVA virtual machine and virtual 8086 mode, hard and soft real time operating system, pre-emptive and non-pre-emptive multitasking and scheduling inter process communication shared memory semaphore message queues, signals sessions management, multiprocessor and distributed process synchronization, symmetric multiprocessing systems.

UNIT-III
Advanced Memory Management

Virtual address space, description of user process and kernal, virtual memory architecture of Pentium group of processor. Translation Lookaside Buffers, implementation of file mapping, shared memory through virtual memory virtual swap space.

UNIT - IV
Advanced Device Management Feature

Device driver framework classifying devices and driver, invoking driver code, devices switch table and driver entry points, dynamic loading and unloading of device drivers

UNIT V
Advanced File Management Features

Virtual file systems and v-node architecture, distributed file system, network file system, remote procedure call

RECOMMENDED BOOKS
1. Principles of Operating System   -Peterson.
2. Operating System   - Mandinick & Donovan.
Suggested Digital Platforms Web Links:
https://onlinecourses.nptel.ac.in/noc22_cs78/preview
https://onlinecourses.nptel.ac.in/noc22_cs100/preview

Paper III: Data Structure through algorithms using 'C'

Max Marks:100   **Min Marks:40**

NOTE: - The Question Paper setter is advised to prepare unit-wise question with the provision of internalchoice.

Course Outcome:

Students will be able to:

- Develop efficient algorithms for solving a problem.
- Use the various construct of programming language viz., conditional, iteration and recursion.
- Implement the algorithm in programming language.
- Use simple data structure like array, stacks and linked list in solving problems.

UNIT – I: Introduction and Preliminaries -

Introduction, Basic terminology, Elementary data organization, Data structure, Data structure operation, Algorithms:complexity, time-space Tradeoff.. Mathematical Notation and functions, Algorithmic Notation, Control Structures, Complexity of Algorithms, Sub algorithms, Variables, Data Type.

UNIT - II:String Processing, Arrays, Records and Pointers –

Basic Terminology, Storing String, Character Data Type, String Operations, Word Processing, Pattern Matching Algorithms. Linear Array, Representation of linear Array in Memory, Traversing Linear Arrays, Inserting And Deleting, Sorting; Bubble Sort, Searching; Liner Search, Binary Search, Multidimensional Array, Pointers; Pointer Array, Records; Record Structures, Representation of Records in Memory; Parallel Arrays, Matrices, SparseMatrices.

UNIT - III: Linked Lists, Stacks, Queues, Recursion -

Linked list, Representation of linked lists in memory, Traversing a linked list, Searching a linked list, Memory Allocation; Garbage Collection, Insertion into a linked List, Deletion from a Linked List, Header Linked List, Two- Way Linked Lists. Stacks, Array Representation of Stack, Arithmetic Expressions; Polish Notation, Quick sort, an application of Stacks, Recursion, Tower of Hanoi, Implementation of Recursive Procedures by Stacks, Queues, Dequeues, Priority Queues.

UNIT - IV : Trees & Graphs -

Binary Trees, Representing Binary Trees in Memory, Traversing binary tree, Traversal Algorithms using stacks, header nodes; threads, Binary Search Tree, Searching and Inserting in Binary Search Tree, Deleting in Binary Search tree, Heap; Heap sort, Path Lengths; Huffmans Algorithms, General Tree. Graph Theory Terminology, Sequential Representation of Graph; Adjacency Matrix, Path Matrix, Linked Representation of Graph.

UNIT - V : Sorting And Searching –

Sorting, Insertion Sort, Selection Sort, Merging, Merge Sort, Radix Sort, Searching and data modification, hashing.

BOOKS RECOMMENDED :

1. *DataStructure*   - **Seymour Lipschutz (Schaum'sSeries).**
2. *Data Structure &ProgramDesign*   - **Robert L. Kruse, 3rdEd., PrenticeHall.**

Suggested Digital Platforms Web Links:

https://onlinecourses.nptel.ac.in/noc22_cs92/preview
https://onlinecourses.swayam2.ac.in/cec22_cs19/preview
https://onlinecourses.nptel.ac.in/noc22_cs70/preview

Paper IV : Object Oriented Programming using 'C++'

Max Marks:100   **Min Marks:40**

NOTE: - The Question Paper setter is advised to prepare unit-wise question with the provision of internalchoice.

Course Outcome:
Students will be able to:

- Understand object oriented programming, difference between object oriented programming and procedural programming.
- Able to build program using C++ features such as Class, objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.
- Able to build C++classes using appropriate encapsulation and design principles.
- Improve problem solving skills by applying object oriented or non-object oriented techniques

UNIT - I : Language Fundamental
Advantages of OOP, The Object Oriented Approach, and Characteristics of object oriented languages-Object, Classes, Inheritance, Reusability, and Polymorphism.
Overview of C++: History of C++, Data Types, Constants and Variables, Operators and Expression. Control structures : if , if-else, nested if-else, while(), do-while() , for(;;), break, continue, switch, goto, String, Storage class.

UNIT - II : Structure, Function & Array
Structures : A Simple structures ,specify the structures, Defining a structure variable, Accessing structures member, Enumeration data type.
Function: Function Declaration, Calling Function, Function Defines, Passing Argument to function, Passing Constant, Passing Value, Reference Argument, Passing struct variable, Overloaded Function, Inline Function, Default Argument, return statement, returning by reference.
Array: Defining array, array element, initiation array, multi dimensional array, passing array to function.

UNIT - III : Object Classes and Inheritance
Object and Class, Using the class, class construct, class destructors, object as function argument, struct andclasses, array as class member, operator overloading. Type of inheritance, Derive class, Base class. Access specifier: protected. Overriding, memberfunction.

UNIT - IV : Pointers
Pointers : & and * operator pointer variables, pointer to void ,pointer and array, pointer and function, pointer and string, memory management, new and delete, pointer to object, pointer to pointer.

UNIT - V : Virtual Function and File & Stream
Virtual Function : Virtual Function, Virtual member function, accesses with pointer, Late binding, pure virtual function, Friend function, Friend class, static function, this pointer, Templates.

File and Stream: C++ streams, Stream class, string I/O, char I/O, Object I/O, I/O with multiple object, File pointer, Disk I/O.

RECOMMENDED BOOKS :

1. ObjectOrientedProgramming   : McGregor and Sykes S A, 1992 VanNostrand.
2. The C++ProgrammingLanguage   : StrustrpB,AddisionWasley.
3. Object Oriented ProgramminginC++   : Lafore R, GalgotiaPublications.
4. Introduction to Object Oriented Programming : Witt KV, GalgotiaPublications.
5. ObjectOrientedProgramming   : Blaschek G, SpringerVerlag
6. ObjectDataManagement   : Cattel R, AddisonWasley.

Suggested Digital Platforms Web Links:
https://onlinecourses.swayam2.ac.in/aic20_sp01/preview

Paper V: Computer System Architecture

MaxMarks:100   **Min Marks:40**

NOTE:- The Question Paper setter is advised to prepare unit-wise question with the provision of internalchoice.

Course Outcome:
Students will be able to:
- Get concepts of the basics organizational and architectural issues of a digital computer.
- Analyze performance issues in processor and memory design of a digital computer.
- Understand various data transfer techniques in digital computer.
- Explain block diagram of CPU, Memory and types of I/O transfers

UNIT - I : Representation of Information
Number system, Integer & Floating point representation Character code (ASCII, EBCDIC), Error Detect and Correct code, Basic Building Blocks, Boolean Algebra, MAP Simplification, Combination Blocks, Gates, Multiplexers, Decoders, etc Sequential building block, flip-flop, registers, counters, ALU, RAM etc.

UNIT - II : Register transfer language and micro operations
Concepts of bus, data movement along registers, a language to represent conditional data transfer, data movement from its memory, arithmetic and logical operations along with register transfer timing in register transfer

UNIT - III : Basic Computer Organization and Design
Instruction code, Computer Instructions, Timing and Control, Execution of Instruction, Input and Output Interrupt, Design of Computer.

UNIT - IV : Computer Software
Programming Language, Assembly Language, Assembler, Program Loops, Input /Output Programming, System Software. Central Processor Organization: - Processor Bus Organization, Arithmetic Logic Unit, Stack Organization, Instruction Formats, Addressing modes, Data transfer and Manipulation, Program Control, Microprocessor Organization, Parallel Processing,.

UNIT - V : Input –Output & Memory Organization
Input –Output Organization : Peripheral Devices, Input/Output Interface, Asynchronous Data Transfer, Direct Memory Access (DMA), Priority Interrupt, Input-Output Processor, Multiprocessor System Organization, and Data Communication Processor.

Memory Organization : Auxiliary Memory, Micro Computer Memory, Memory Hierarchy, Associative Memory, Virtual Memory, Cache Memory, Memory Management Hardware.

BOOKS RECOMMENDED:
1. Computer SystemArchitecture   - M. Morris Mano(PHI).
2. DigitalComputerElectronics   -Malvino.
3. Digital Computers andLogicDesign   - M.Morris Mano(PHI).
4. StructuredComputerOrganization   - Andrew M. Tanenbanm(PHI).

Suggested Digital Platforms Web Links:
https://onlinecourses.nptel.ac.in/noc22_cs88/preview
https://onlinecourses.nptel.ac.in/noc22_cs110/preview

Practical

M.Sc. (CS) - I: Practical Based on Paper III (Data Structure through Algorithms using 'C')

Course Outcome:

Students will be able to:

- Implement efficient algorithms and identity the appropriate data structure for solving a problem using C.
- Implement simple data structure like array and stacks using C.
- Be able to design and analyze the time and space efficiency of the data structure ·
- Have practical knowledge on the applications of data structures.
- Implement abstract data types using arrays and linked list.
- Implement different types of trees and apply them to problem solutions.
- Discuss graph structure and understand various operations on graphs and their applicability.
- Analyze the various sorting and searching algorithms.

SchemeofExamination: -

Practicalexaminationwillbetwoprogramsandaprojectdemonstration.Itwillbeofthreehours duration.Allprogrammewithflowchartandalgorithms.Thedistributionofpracticalmarks willbeasfollows

Programme1	-	20
Programme2	-	20
Programme3	-	20
Viva	-	25
[PracticalCopy+		
InternalRecord]	-	15
Total	-	**100**

Practical List

Note: This is tentative list; the teachers concern can add more program as per requirement.

1. Write an algorithm and program to demonstrate the concept of one-dimensional array to finding the sum of array elements.
2. Write an algorithm and program to insert an element in an array.
3. Write an algorithm and program to delete an element from an array.
4. Write an algorithm and program to add two matrix A and B.
5. Write an algorithm and program to multiply two matrix A and B.
6. Write an algorithm and program to demonstrate pointer and an array.
7. Write an algorithm and program to Implementation of linked list using array.
8. Write an algorithm and program to insert an item into double linked list.
9. Write an algorithm and program to delete an item from double linked list.
10. Write an algorithm and program to Implementation of stack using array.
11. Write an algorithm and program to Implementation of queue using array.
12. Write an algorithm and program to Implementation of circular queue using array.
13. Write an algorithm and program to Implementation of binary search tree using array.
14. Write an algorithm and program to search an element using sequential search.
15. Write an algorithm and program to search an element using binary search.
16. Write an algorithm and program to arrange the list of numbers in ascending order using Bubble Sort.

17. Write an algorithm and program to arrange the list of numbers in ascending order using Insertion Sort.
18. Write an algorithm and program to arrange the list of numbers in ascending order using Selection Sort.
19. Write an algorithm and program to arrange the list of numbers in ascending order using Merge Sort.
20. Write an algorithm and program to arrange the list of numbers in ascending order using Quick Sort.
21. Write an algorithm and program to arrange the list of numbers in ascending order using Radix Sort.
22. Write an algorithm and program to arrange the list of numbers in ascending order using Heap Sort.
23. Write an algorithm and program to demonstrate DFS.
24. Write an algorithm and program to demonstrate BFS.
25. Write an algorithm and program to demonstrate string manipulation functions.
26. Write an algorithm and program to covert infix expression to prefix expression.
27. Write an algorithm and program to covert infix expression to postfix expression.
28. Write an algorithm and program to demonstrate In-order Traversal in tree.
29. Write an algorithm and program to demonstrate Pre-order Traversal in tree.
30. Write an algorithm and program to demonstrate Post-order Traversalin tree.

M.Sc. (CS) - I : Practical Based on Paper IV(Object Oriented Programming Using 'C++ ')

Course Outcome:

Students will be able to:

- Able to build program using C++ features such as Class, objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.
- Able to build C++ classes using appropriate encapsulation and design principles.

SchemeofExamination:-

Practicalexaminationwillbetwoprogramsandaprojectdemonstration.Itwillbeof3hours duration.Allprogrammewithflowchart&algorithms.Thedistributionofpracticalmarks willbeasfollows

Programme1	-	20
Programme2	-	20
Programme3	-	20
Viva	-	25
[PracticalCopy+		
InternalRecord]	-	15
Total	-	**100**

Practical List

Note: This is tentative list; the teachers concern can add more program as per requirement.

1. Create a class student having data members to store roll number, name of student, name of three subjects, max marks, min marks, obtained marks. Declare array of object to hold data of 3 students. Provide facilities to display result of all students. Provide also facility to display result of specific student whose roll number is given.

2. Write a Program using class to process Shopping List for a Departmental Store. The list includes details such as the Code No and Price of each item and perform the operations like Adding, Deleting Items to the list and Printing the Total value of an Order.

3. WAP in C++ to reverse the string without using library functions.

4. WAP in C++ to count the number of characters in string passed as argument in form of character array.

5. Write a Program to design a class having static member function named showcount() which has the property of displaying the number of objects created of the class

6. Create a class Simple with static member functions to find factorial by recursive member function

7. Write a Program which creates & uses array of object of a class. (for e.g. implementing the list of Managers of a Company having details such as Name, Age, etc.)

8. Write a program to create class Mother having data member to store salary of Mother, create another class Father having data member to store salary of Father. Write a friend function, which accepts objects of class Mother, and Father and prints Sum of Salary of Mother and Father objects.

9. Write a Program to find Maximum out of Two Numbers using friend function.

10. Write a Program to swap private data members of classes named as class_1, class_2 using friend function.

11. Write a program using inline function to find minimum of two functions. The inline function should take two arguments and should return the minimum value

12. Write a Program to design a class complex to represent complex numbers, the complex class should use an external function (use it as a friend function) to add two complex numbers. The function should return an object of type complex representing the sum of two complex numbers.

13. Write a Program using copy constructor to copy data of an object to another object

14. Write a Program to allocate memory dynamically for an object of a given class using class's constructor.

15. Write a Program to design a class to represent a matrix. The class should have the functionality to insert and retrieve the elements of the matrix

16. Write a program to design a class representing complex numbers and having the functionality of performing addition & multiplication of two complex numbers using operator overloading.

17. Write a program for developing a matrix class which can handle integer matrices of different dimensions. Also overload the operator for addition, multiplication & comparison of matrices

18. Write a program to overload new/delete operators in a class.

19. Write a program in C++ to highlight the difference between overloaded assignment operator and copy constructor.

20. write program to create a class Polar which has data member radius and angle, define overloaded constructor to initialize object and copy constructor to initialize one object by another existing object keep name of parameter of parameterized constructor same as data members. Test function of the program in main function.

21. write program to create a class Polar which has data member radius and angle, use constructor with default arguments to avoid constructor overloading and copy constructor to initialize one object by another existing object keep name of parameter of parameterized constructor same as data members. Test functioning of the program in main function

22. Write a program to create a class Employee having data members to store name of employee, employee id, salary. Provide member function for data input, output. Use Pointer to object to simulate array of object to store information of 3 employees and test the program in function main.

23. Write a Program illustrating how the constructors are implemented and the order in which they are called when the classes are inherited. Use three classes named alpha, beta, gamma such that alpha, beta are base class and gamma is derived class inheriting alpha & beta

24. Write a Program to design a student class representing student roll no. and a test class (derived class of student) representing the scores of the student in various subjects and sports class representing the score in sports. The sports and test class should be inherited by a result class having the functionality to add the scores and display the final result for a student

25. Write a program to maintain the records of person with details (Name and Age) and find the eldest among them. The program must use this pointer to return the result.

26. Write a Program to illustrate the use of pointers to objects which are related by inheritance.

27. Write a program illustrating the use of virtual functions in class.

28. Create a base class shape having two data members with two member function getdata (pure virtual function) and printarea (not pure virtual function). Derive classes triangle and rectangle from class shape and redefine member functioning of classes using pointer to base class objects and normal objects.

29. Write a program to design a class representing the information regarding digital library (books, tape, book & tape should be separate classes having the base class as media). The class should have the functionality for adding new item, issuing, deposit etc. the program should use the runtime polymorphism

30. Write a program implementing basic operation of class ios i.e. setf, unsetf, precision etc.

31. Write a program to implement I/O operations on characters. VO operations includes inputting a string. Calculating length of the string, Storing the string in a file, fetching the stored characters from it, etc.

32. Write a program to copy the contents of one file to another.

33. Write a program to perform read/write binary I/O operation on a file (i.e. write the object of a structure/class to file).

34. Write a program for reading and writing data to and from the file using command line arguments.

35. Write swapping program to demonstrate call by value, call by address and call by reference in a single program

MASTER OF SCIENCE IN COMPUTER SCIENCE

SECOND SEMESTER

Subject Code	SUBJECTS	Teaching Load Per			Credit L+(T+P)/2	Examination Marks							
						Max. Marks				Min. Marks			
		L	T	P		Th	Ses	Pr	Total	Th	Ses	Pr	Total
Paper 1	RDBMS (SQL Programming with Oracle)	3	2	-	4	100	50	-	150	40	30		70
Paper II	Advanced Computer Networks	3	2	-	4	100	50	-	150	40	30		70
Paper III	Python Programming	3	2	-	4	100	50	-	150	40	30		70
Paper IV	Principles of Compiler Design	3	2	-	4	100	50	-	150	40	30		70
Paper V	Numerical Analysis	3	2	-	4	100	50	-	150	40	30		70
Practical I	Practical Based on Paper I			3x2	3		25	100	125			15	50	65
Practical II	Practical Based on Paper III	-	-	3x2	3	-	25	100	125	-	15	50	65
TOTAL		1	10	12	2	500	30	20	1000	20	18	10	480

Paper I: RDBMS (SQL Programming with Oracle)

Max Marks:100   **Min Marks :40**

NOTE :- The Question Paper setter is advised to prepare unit-wise question with the provision of internalchoice.

Course Outcome
Students will be able to:

- Establish a basic understanding of the process of Database Development and Administration using MySQL.
- Student will implement the concepts of both Operating Systems & Database Administration skills.
- Understand fundamental concepts of RDBMS (SQL/PLSQL)
- Understand functioning of database management systems as well as associated tools and techniques
- Develop a good database design and normalization techniques to normalize a database.
- Able to write Procedure, Function, Cursor and Trigger using SQL/PLSQL.

UNIT - I : Overview of Database Management -
Data, Information and knowledge, Increasing use of data as a corporate resource, data processing verses data management, file oriented approach verses database oriented approach to data management; data independence, database administration roles, DBMS architecture, different kinds of DBMS users, importance of data dictionary, contents of data dictionary, types of database languages. Data models: network, hierarchical, relational. Introduction to distributed databases, Client/Server databases, Object-oriented databases, Object-relational databases, Introduction to ODBC concept.

UNIT - II : Relational Model & Relational Algebra -
Entity - Relationship model as a tool for conceptual design-entities, attributes and relationships. ER diagrams; Concept of keys; Case studies of ER modeling Generalization; specialization and aggregation. Converting an ER model into relational Schema. Extended ER features, Introduction to UML, Representation in UML diagram (Class Diagram etc.).
Relational Algebra: select, project, cross product different types of joins (inner join, outer joins, self join); set operations, Tuple relational calculus, Domain relational calculus, Simple and complex queries using relational algebra, stand alone and embedded query languages.

UNIT - III :SQL
Introduction to SQL constructs (SELECT…FROM, WHERE… GROUP BY… HAVING… ORDERBY….), INSERT, DELETE, UPDATE, DROP, VIEW definition and use, Temporary tables, Nested queries, and correlated nested queries, Integrity constraints: Not null, unique, check, primary key, foreign key, references, Triggers. Embedded SQL and Application Programming Interfaces.
Introduction to PL/SQL variables – literals – data types – advantages of PL/SQL; Control statements:if ; iterative control – loop, while, for , goto ; exit when; Cursors : Types – implicit, explicit – parameterized cursors – cursor attributes; Exceptions: Types – internal , user-defined , handling exceptions – raise statement.

UNIT - IV :PL/SQL
PL/SQL tables and records: Declaring PL/SQL tables - referring PL/SQL tables, inserting and fetching rows using PL/SQL table, deleting rows; records - declaration of records - deleting records; Sub programs: Functions - procedures – input-output parameters; purity functions - packages - package specification - advantages of packages - private and public items - cursors in packages.

UNIT - V : Relational Database Design-

Normalization concept in logical model; Pitfalls in database design, update anomalies: Functional dependencies, Join dependencies, Normal forms (1NF, 2NF, 3NF). Boyce Codd Normal form, Decomposition, Multi-Valued Dependencies, 4NF, 5NF. Issues in physical design; Concepts of De-normalization, Indexing, Clustering indexes. Data Organization - Fixed length records, variable length records, Organization of records in files, Indexing: - indexed files -B-tree, B+-tree, and Hashing Techniques.

BOOKS RECOMMENDED :

1. Databasesystemconcept   - **H. Korth and A. Silberschatz,TMH**
2. Data BaseManagementSystem   - **IvanBayross**
3. Data BaseManagementSystem   - **JamesMatin**
4. DatabaseManagementSystem   - **Leon & Leon, VikasPublication**
5. An Introduction todatabasesystems   - **Bipin Desai, GalgotiaPublication.**
6. DatabaseManagementSystem   - **A. K. Majumdar &P.Bhattacharya,TMH**

Suggested Digital Platforms Web Links:

Paper II: Advanced Computer Networks

Max Marks:100   **Min Marks :40**

NOTE :- The Question Paper setter is advised to prepare unit-wise question with the provision of internalchoice.

Course Outcome
Students will be able to:
- Understand basic computer network technology.
- Understand Data Communications System and its components.
- Enumerate the layers of the OSI model and TCP/IP reference model.
- Able to identify the different types of network devices, their functions within a network and their applications.

UNIT - I
Introduction to Computer Networking : The Concept of Networking, Data Communication, Required network elements, The role of Standards Organization. Line Configuration, Various Topologies, Transmission Mode, Categories of Networks- LAN, MAN, WAN. The benefits of a Computer Networks.

The OSI and TCP/IP Reference Model : The Concept of Layered Architecture, Design Issues for the Layers. Interfaces and services, Detailed Functions of the Layers. Comparison between OSI and TCP/IP Reference model.

UNIT - II
Transmission of Digital Data : Shannon's and Nyquist theorems for maximum data rate of a channel. Transmission media- Co-axial, UTP, Fiber optic and wireless. Analog and digital data Transmission- parallel and serial transmission. DTE-DCE interface using RS-232C. Study of modems- 56k and Cable Modem. Modem standards.

Multiplexing and Switching : The Concept of Multiplexing- FDM, TDM, WDM. The Concept of Switching- Circuiting, Message switching, Packet switching.

UNIT - III
Data Link Layer and Routing Algorithms : Line Discipline, Flow Control- stop and wait, sliding window, Go back N, Error Control- ARQ stop and wait, sliding window ARQ. HDLC, SLIP, PPP. Multiple access protocols- ALOHA, Slotted ALOHA, CSMA/CD. IEEE standards for LAN's and MAN's. The IP protocol, and its header. IP address classes and subnetmask.

The concept of ICMP, ARP, RARP, RSVP, CIDR and Ipv6. : Routing algorithms- shorted path first, Distance Vector, Link State. Congestion Control-The leaky bucket and Token bucket Algorithms.

UNIT - IV
Transport Layer : The Concept of client and Server in terms of Socket addressing in Transport layer. Two way and three-way handshaking. TCP header.
Network Performance Issues. The Concept of Domain Name System, Various Resource Records. Architecture and services of E-mail (RFC-822 and MIME). The Concept of World Wide Web- server side and client side.

ATM : The concept of ATM, ATM Adoption layers- AAL1, AAL2, AAL3/4, AAL5, Comparison of AAL protocols. Cell formats for UNI and NNI. Service Categories, Quality of service, Congestion Control in ATM.

UNIT - V
Comparative study of Networking Technologies: X.25, Frame Relay, ATM, SONET, SMDS, ISDN.
Network Security: The importance of Security in Networking, traditional cryptography, Data Encryption standards, RSA Algorithm.

BOOKS RECOMMENDED :
ComputerNetworks   - A S Tanenbaum
Data CommunicationandNetworking   -Forouzan

Suggested Digital Platforms Web Links:
https://onlinecourses.nptel.ac.in/noc22_cs90/preview
https://onlinecourses.nptel.ac.in/noc22_cs112/preview
https://onlinecourses.nptel.ac.in/noc22_cs114/preview

Paper III: Python Programming

Max Marks:100   Min Marks :40

NOTE :- The Question Paper setter is advised to prepare unit-wise question with the provision of internal choice.

Course Outcome

Students will be able to:
- Knowledge about the Python programming.
- Define the Structure and Components of a Python Program.
- Demonstrate proficiency in handling of loops and creation of functions. Identify the methods to create
  and manipulate lists, tuples and dictionaries.
- Discover the commonly used operations involving regular expressions and file systems.
- Determine the need of scrapping website and working with CSV, JSON and other file formats.
- Discover the commonly used of data science in Python.
- Demonstrate proficiency in handling of NumPY & Pandas Library.
- Demonstrate proficiency in handling of GUI programming using Tkinter.

Unit I

Introduction to Python :- Structure of a Python Program, Elements of Python. Python Interpreter ,Installing Python, basic syntax, interactive shell, editing saving and running a script; The concept of data types, variables, assignments; immutable variables; numerical types, operators(Arithmetic Operator, Relational Operator, Logical or Boolean Operator, Assignment Operator, Ternary Operator, Bitwise Operator, Increment or Decrement Operator) and expressions; comments in the program,

Unit II

Creating Python Programs: Input and Output Statements, Control Statements (Branching, Looping, Conditional Statement, Exit function, Difference between break, continue and pass). Function : Defining a function, calling a function, types of function, Function Arguments, Anonymous Functions, global and local variables, Recursion , Strings and Text Files: - Manipulating files and directories, os and sys modules, text files: reading/writing text and numbers from/to a file, creating and deleting a formatted file (csv or tab-separated).

Unit III

Lists, Tuples and Dictionaries :Basic list operators, replacing, inserting and removing an element, searching and sorting lists, Accessing tuples, Operations, Working Functions and Methods, dictionary literals, Adding and Removing keys, accessing and replacing values, traversing dictionaries. Package-Introduction, importing from package, json, Exception handling - try,else, finally, raise keyword.

Unit IV

Pandas Introduction, installation of panda, data frame, series, range data, slice data,drop a column, concatenation. Data science using python, Data Frame-Creating Data Frame from an Excel Spreadsheet, Creating Data Frame from .csv file, Creating Data Frame from python Dictionary, Creating Data Frame from python List of Tuples, Operations on Data Frames. Data visualization using Matplot Lib Bar Graph ,Histogram ,Creating a pie chart ,creating line graph
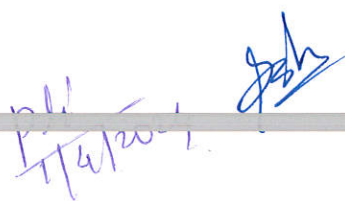
Unit V

NumPY Introduction, creating NumPY arrays, indexing and slicing in NumPy. GUI Programming: Introduction to GUI library, Advantages, Layout management, Events and binding Drawing on canvas(line, oval, rectangle etc) widget such as Frame, Label, Button, Checkbutton, Entry, Listbox, Radiobutton, Text, Spinbox Tkinter introduction, Tkinter and Python Programming, Tk Widgets widget such as Frame, Label, Button, Checkbutton, Entry, Listbox, Radiobutton, Text, Spinbox

BOOKS RECOMMENDED:

1. T. Budd, Exploring Python, TMH, 1st Ed, 2011
2. Allen Downey, Jeffrey Elkner, Chris Meyers, How to think like a computer scientist: Learning with Python, Freely available online, 2012
3. Luca Massaron John Paul Mueller, Python for Data science For Dummies, Wiley,2ed,2019
4. Introduction to computation and programming using python: with application to understanding data, John V. Guttag
5. Problem Solving using Python – E. Balagurusamy, Mc Graw Hill Education Ltd., 2017
6. Practical Programming An Introduction to Computer Science Using Python - Jennifer Campbell
7. https://docs.python.org/3/tutorial/index.html
8. http://interactivepython.org/courselib/static/pythonds
9. https://wiki.python.org/moin/TkInter

Paper IV: Principles of Compiler Design

Max Marks: 100   **Min Marks : 40**

NOTE: - The Question Paper setter is advised to prepare unit-wise question with the provision of internal choice.

Course Outcome

Students will be able to:

- Specify and analyze the lexical, syntactic and semantic structures of advanced language features
- Separate the lexical, syntactic and semantic analysis into meaningful phases for a compiler to undertake language translation
- Write a scanner, parser, and semantic analyzer without the aid of automatic generators
- Turn fully processed source code for a novel language into machine code for a novel computer
- Describe techniques for intermediate code and machine code optimization
- Design the structures and support required for compiling advanced language features.

UNIT-I

Introduction to Compilers: Overview, Structure, implementation. Programming Language Grammars: Inter Language grammars, derivation, reduction, syntax tree, ambiguity, regular grammars & expressions.

UNIT-II

Scanning and Parsing Techniques : The Scanner, parser, translation, elementary symbol table organization,structures.

UNIT-III

Memory Allocation: Static and dynamic memory allocation, array allocation and access, allocation for strings, structure allocation, common & equivalence allocation. Introduction to Compilation ofexpressions.

UNIT-IV

Compilation of Control Structures : Control transfers, procedural calls, conditional execution, interation control constructs. Error detection, indication &recovery.
Compilation of I/O Statements: Compilation of I/O list, compilation of FORMAT list, IOSUB, filecontrol.

UNIT-V

Code Optimization: Major issues, optimizing transformations, local optimizations, program flow analysis, Global Optimization, writingcompilers.

BOOKS RECOMMENDED:

1. Compiler Construction -D.M.Dhandhere(M)
2. Compiler Writing -Tremble-Sorenson(TMH)
3. Computers : Princ, Techniques cools by Aho-Person.
4. The Essence of Compilers by Hanter-Pearson.

Paper V: Numerical Analysis

Max Marks: 100   **Min Marks :40**

NOTE: - The Question Paper setter is advised to prepare unit-wise question with the provision of internalchoice.

Course Outcome

Students will be able to:

- Root finding for nonlinear equations,
- Interpolation and approximation of functions by simpler computational building blocks (for example - polynomials and splines).
- Numerical differentiation and divided differences.
- Numerical quadrature and integration,
- numerical solutions of ordinary differential equations and boundary value problems;

UNIT – I : Solution of Polynomial and Transcendental Algebraic Equations

Bisection method, Regulafalsi method & Newton's method, Solution of Cubic & Biquadrate Equation, Complex roots of polynomial equations.

UNIT – II : Simultaneous Equations and Matrix

Gauss-Jordan method, Cholesky's method, Reduction to lower or upper Triangular forms, Inversion of matrix, method of partitioning, Characteristics equation of matrix, Power methods, Eigen values of matrix, Transformation to diagonalforms.

UNIT - III : Curve-Fitting from Observed Data

Divided difference table for evenly or unevenly spaced data, polynomial curve-fitting - Newton's, Gauss and Langranges form of interpolation and Divided Differences, method of least square for polynomials,.

UNIT - IV : Numerical Differentiation and Integration

Forward and Backward differential operators, Newton - cotes integration formula: Trapezoidal Rule, Simpson's Rule, Boole's Rule, Weddle Rule, Legendre's rule, method of weighted coefficients.

UNIT - V : Solution of Differential Equations

Numerical Solution of ordinary differential equations, one step method, Taylor's Series, Predictor-Corrector Method, Euler's Method, Runga-Kutta Method, Milne's method.

BOOKS RECOMMENDED

1. Garewal   :   Numericalmethods
2. Gupta &Mallic :   NumericalMethods
3. Hamming R.W.   :   Numerical methods for scientist & Engineers. (McGrawHill)
4. ConleS.D.   :   Elementary numericalanalysis
   CarlDeBoor   (International Book CompanyLondon)
5. Jain M.K.   :   Numerical methods for ScienceandEngineering
   IyengarS.R.K   Calculations (John Willey &Sons)

M.Sc. (CS) - II: Practical Based on Paper I (SQL Programming with Oracle)

Course Outcome
Students will be able to:
- Establish a basic understanding of the process of Database Development and Administration using MySQL.
- Master the basic concepts and appreciate the applications of database systems.
- Master the basics of SQL and construct queries using SQL.
- Student will implement the concepts of both Operating Systems & Database Administration skills.
- Understand functioning of database management systems as well as associated tools and techniques
- Develop a good database design and normalization techniques to normalize a database.
- Able to write Procedure, Function, Cursor and Trigger using SQL/PLSQL.

SchemeofExamination: -
Practicalexaminationwillbetwoprogramsandaprojectdemonstration.Itwillbeof3hours
duration.Allprogrammewithflowchart&algorithms.Thedistributionofpracticalmarks willbeasfollows

Programme1	-	20
Programme2	-	20
Programme3	-	20
Viva	-	25
[PracticalCopy+ InternalRecord]	-	15
Total	-	**100**

Practical List
Note: This is tentative list; the teachers concern can add more program as per requirement.
SQL

1. Using the following database,
   Colleges (<u>cname</u>, city, address, phone, afdate) Staffs (<u>sid</u>, sname, saddress, contacts)
   StaffJoins (sid, cname, dept, DOJ, post, salary)
   Teachings (sid, class, paperid, fsession, tsession)
   Subjects (<u>paperid</u>, subject, paperno, papername)

Write SQL statements for the following –
a. Create the above tables with the given specifications and constraints.
b. Insert about 10 rows as are appropriate to solve the following queries.
c. List the names of the teachers teaching computer subjects.
d. List the names and cities of all staff working in your college.
e. List the names and cities of all staff working in your college who earn more than15,000
f. Find the staffs whose names start with 'M' or 'R' and ends with 'A' and/or 7 characters long.
g. Find the staffs whose date of joining is2005.
h. Modify the database so that staff N1 now works in C2College.
i. List the names of subjects, which T1 teaches in this session or all sessions.
j. Find the classes that T1 do not teach at present session.
k. Find the colleges who have most number of staffs.
l. Find the staffs that earn a higher salary who earn greater than average salary of their college
m. Find the colleges whose average salary is more than average salary ofC2
n. Find the college that has the smallest payroll.

o. Find the colleges where the total salary is greater than the average salary of all colleges.
p. List maximum, average, minimum salary of each college
q. List the names of the teachers, departments teaching in more than one department.
r. Acquire details of staffs by name in a college or each college.
s. Find the names of staff that earn more than each staff of C2College.
t. Give all principals a 10% rise in salary unless their salary becomes greater than 20,000 in such case give 5%rise.
u. Find all staff that do not work in same cities as the colleges they work.
v. List names of employees in ascending order according to salary who are working in your college or all colleges.
w. Create a view having fields sname, cname, dept, DOJ, andpost
x. Create a view consisting of cname, average salary and total salary of all staff in that college.
y. Select the colleges having highest and lowest average salary using above views.
z. List the staff names of a department using above views.

2. Create the following database,
   Enrollment (enrollno, name, gender, DOB, address, phone)
   Admission (admno, enrollno, course, yearsem, date, cname)
   Colleges (cname, city, address, phone, afdate)
   FeeStructure (course, yearsem, fee)
   Payment (billno, admno, amount, pdate, purpose)
a. Create the above tables with the given specifications and constraints.
b. Insert about 10 rows as are appropriate to solve the following queries.
c. Get full detail of all students who took admission this year class wise
d. Get detail of students who took admission in Bhilai colleges.
e. Calculate the total amount of fees collected in this session
i) By your college
ii) by each college
iii) by all colleges
f. List the students who have not payed full fee
a. in your college ii) in all colleges.
List the number of admissions in your class in every year.
g. List the students in the session who are not in the colleges in the same city as they live in.
h. List the students in colleges in your city and also live in your city.

3. Create the following database,
   Subjects (paperid, subject, paper, papername)
   Test (paperid, date, time, max, min)
   Score (rollno, paperid, marks, attendence)
   Students (admno, rollno, class, yearsem)
a. Create the above tables with the given specifications and constraints.
b. Insert about 10 rows as are appropriate to solve the following queries.
c. List the students who were present in a paper of a subject.
d. List all roll numbers who have passed in first division.
e. List all students in BCA-II who have scored higher than average

i) in your college ii) in every college
f. List the highest score, average and minimum score in BCA-II
   i) in your college ii)in every college

4. Using the following database
   Colleges (<u>cname</u>, city, address, phone,afdate)
   Staffs (<u>sid</u>, sname, saddress, contacts)
   StaffJoins (sid, cname, dept, DOJ, post, salary)
   Teachings (sid, class, paperid, fsession, tsession)
   Subjects (<u>paperid</u>, subject, paperno, papername)

Write SQL statements for the following –
a. Create the above tables with the given specifications and constraints.
b. Insert about 10 rows as are appropriate to solve the following queries.
c. List the names of the teachers teaching computer subjects.
d. List the names and cities of all staff working in your college.
e. List the names and cities of all staff working in your college who earn more than15,000

5. Using the following database
   Colleges (<u>cname</u>, city, address, phone, afdate)
   Staffs (<u>sid</u>, sname, saddress, contacts)
   StaffJoins (sid, cname, dept, DOJ, post, salary)
   Teachings (sid, class, paperid, fsession, tsession)
   Subjects (<u>paperid</u>, subject, paperno, papername)

a. Find the staffs whose names start with 'M' or 'R' and ends with 'A' and/or 7 characters long.
b. Find the staffs whose date of joining is2005.
c. Modify the database so that staff N1 now works in C2college.
d. List the names of subjects which T1 teaches in this session or all sessions.

6. Using the following database
Colleges (<u>cname</u>, city, address, phone, afdate) Staffs (<u>sid</u>, sname, saddress, contacts)
   StaffJoins (sid, cname, dept, DOJ, post, salary)
Teachings (sid, class, paperid, fsession, tsession) Subjects (<u>paperid</u>, subject, paperno, papername)

a. Find the classes that T1 do not teach at present session.
b. Find the college who have most number of staffs.
c. Find the staffs who earn a higher salary who earn greater than average salary of their college.
d. Find the colleges whose average salary is more than average salary ofC2
e. Find the college that has the smallest payroll.
f. Find the colleges where the total salary is greater than the average salary of all colleges.
g. List maximum, average, minimum salary of each college