GAUHATI UNIVERSITY

TDC GENERAL COURSE IN COMPUTER SCIENCE (SEMESTER BASED)

Eligibility: Higher Secondary (Science) pass with Mathematics as one of the subjects at the Higher Secondary level securing minimum pass mark in the subject..

Semester - 1

Paper Code	Paper Name	Marks	Classes / week	Credit
E101	Fundamentals of Computer Science	75 marks	6 (5 lectures + 1 tutorial)	6

Semester - 2

Paper Code	Paper Name	Marks	Classes / week	Credit
E201	Introduction to Programming in C	75 marks	6 (5 lectures + 1 tutorial)	6

Semester - 3

Paper Code	Paper Name / Topics	Marks	Classes / week	Credit
E301	Data Structure	50 marks	4 (3 lectures + 1 tutorial)	4
E302	Practical Programming in C Data Structure	Total 50 marks 25 marks 25 marks	4 laboratory sessions	4

Semester - 4

Paper Code	Paper Name / Topics	Marks	Classes / week	Credit
E401	Introduction to Database Management System	50 marks	4 (3 lectures + 1 tutorial)	4
E402	Practical DBMS	50 marks	4 laboratory sessions	4

Semester - 5

Paper Code	Paper Name / Topics	Marks	Classes / week	Credit
E501	Computer Organization and Operating System	100 marks	8 (6 lectures + 2 tutorial)	8
E502	Practical Operating System ICT Hardware	Total 100 marks 40 marks 60 marks	8 (2 tutorials and 6 practical sessions)	8

Semester - 6

Paper Code	Paper Name / Topics	Marks	Classes / week	Credit
E601	Object Oriented Programming in C++ and Computer Networks	100 marks	8 (6 lectures + 2 tutorial)	8
E602	Practical OOP Computer Networks	Total 100 marks 70 marks 30 marks	8 (2 tutorials and 6 practical sessions)	8

DETAILED SYLLABUS
E101 FUNDAMENTALS OF COMPUTER SCIENCE

Total Marks: 75 (Internal 15 and semester end examination 60)
Fundamentals **marks: 8**
Major components of a computer (A brief introduction of CPU, main memory, I/O units). Keyboard, display, mouse, printers etc. Secondary storage devices (hard disks, optical disks, flash memory), backup devices .Computer Software.

Number System: **marks: 7**
Brief description of Binary, Octal, Decimal & Hexadecimal numbers. Characters, ASCII, EBCDIC, BCD, Gray code. Conversion of bases. Complement notations. Representation of signed integers (sign and magnitude, 1's complement, 2's complement) binary arithmetic on signed and unsigned integers.

Boolean Algebra and Logic gates: **marks: 12**
Boolean operators. Rules (postulates & basic theorems) of Boolean algebra. De Morgan's Law. Dual & complement of Boolean expression, Sum of products & product of sum forms. Conversion between different forms (from SOP to POS and vice-versa) .Truth table, logic gates. Simplification of Boolean expressions using algebraic methods. Realization of logic expressions using gates and simplification of Boolean expressions using K-map.

Algorithms: **marks: 12**
Concept of algorithm and flow chart. Writing simple algorithms and drawing flow charts for simple problems like finding sum, max, min, average of a list of numbers etc.

OS & Internet: **marks: 6**
What is an operating system? Evolution of OS, types of OS, functions of OS. Bootstrapping. DOS- Internal & external commands.

Networks: **marks: 10**
Introduction to networking, components of a network, advantages of networking .Basic features LAN ,MAN, WAN, and Wireless LAN

Internet **marks: 5**
Evolution of internet, Basic internet terms(Client, Server, MODEM, Web page, Web site, Home page, Browser, URL, ISP, Web server, Download & Upload, Online & Offline), Internet applications(e-mail, search engines, ftp, VOIP, Video Conferencing, Audio-Video streaming, Chatting).

Suggested Readings:

1. Computing Fundamentals and C Programming; E. Balaguruswamy; Tata McGraw Hill
2. Computer Fundamentals, Anita Goel, Pearson, 2010.

E201 INTRODUCTION TO PROGRAMMING IN C

Total Marks: 75 (Internal 15 and semester end examination 60)

Unit - 1

Fundamentals of C marks: 20

Elementary data types, variables, constants and identifiers. Integer, character floating point and string constants. Variable declarations. Syntax and semantics. Reserved words. Initialization of variable during declaration. Constant data types. Expressions in C- Operator precedence and associativity. Unary, binary and ternary operators. C arithmetic operators, assignment operators, relational operators, logical operators and bit-wise operators. L-value and R-value. Side effects of operators. Expression statement.

Conditional statements – if, if – else, switch
Iterative statements –while, do –while, for
Other statements- break, continue, goto, return, null statement , block statement.
Simple programs like programs to compute – an arithmetic expression, unit conversion, the sum of a series (like trigonometrical series), gcd, factorial, fibonacci number, generation of prime numbers, reversing digits of an integer, finding the square root of a number, generation of pseudo random numbers, prime factors of an integer, base conversion of numbers. Test if three points form a triangle and classify the triangle to right angled, isosceles, equilateral etc. Roots of a quadratic equation. Generating simple patterns of characters on screen.

Unit – 2

Functions marks: 10

Declaration and definition of functions. Calling a function. Parameters- call by value, call by reference and its absence in C.

Cast and size of operator. Automatic type conversion.

Unit – 3 marks: 20

Arrays and Pointers

Different types of variables- local, global, register, static extern. Scope and lifetime of variable.

Arrays and pointers and corresponding operators. Pointer arithmetic.

Programs using arrays and pointers like sum, average, minimum, maximum of an array of numbers. Add and delete an element of an array. Merge two sorted arrays. String manipulation programs-like reverse, test for palindrom, copy, concatenate, find length etc. Matrix manipulation programs like addition, subtraction, multiplication and their combinations. Sum rows, columns, and diagonal elements of a matrix. Transpose a matrix.

Selection and bubble sort. Linear search and binary search.

Unit – 4 **marks: 10**

Structures

Structure –Declaration and use. Structure de reference and structure pointer dereference operators.

Programs to show the use of structures.

Array of structure.

Standard C library.

Suggested Readings:

1. Programming with C, B.S. Gottfried, Tata Mc-Graw Hill.
2. Programming in ANSI C, E.Balagurusamy, Tata McGraw – Hill
3. The C Programming Language, B.W. Kernighan and D.M.Ritchie, PHI

E301DATA STRUCTURE

Total Marks: 50 (Internal 10 and semester end examination 40)

Arrays: types, memory representation, address translation functions for 1 and 2 dimensional arrays; different examples**.** Dynamic storage management. **marks: 3**

Linked structures: singly and doubly linked list (non circular and circular); list manipulation with pointers; examples involving insertion and deletion of elements and their comparative study with implementation using array structure. **marks: 6**

Stacks and Queues: Definitions, representation using array and linked list structure; application of stack & queue in postfix conversion and evaluation of arithmetic expressions. m**arks: 5**

Binary trees: Definitions, quantitative properties; memory representations. Tree traversal algorithms-recursive versions of in order, pre order and post order traversal. **marks: 10**

Searching: Linear and binary search algorithms-performance and complexity. Binary search trees-construction, insertion, deletion and search. Optimal binary search trees (Concept only). **marks: 8**

Sorting: Terminology; performance evaluation. Sorting techniques [complexity, advantages and disadvantages, implementations]-Selection, Bubble, Insertion, Quicksort and Mergesort. **marks: 8**

Suggested reading:

1) Data Structure , Horowitz and Sahani, Narosa
2) Introduction to Data Structures in C, A.N.Kamthane, Pearson, 2007.
3) Data Structure using C and C++, Langsam**,** Augentein & Tanenbaum, PHI
4) Data Structures using C, S.K.Bandyopadhyay, K.N.Dey, Pearson.

E302 PRACTICAL

Part: I Programming in C

Total marks: 25 (Internal 5 and semester end examination 20)

At least 15 programming assignments have to be done by each student from the following list. The assignments should be selected in such a way that all the features of C language are included.

.

1. Write a program to convert a given temperature value from Fahrenheit scale to Centigrade scale and vice versa.
2. Write a program to display ASCII value of a character.
3. Write a program to check whether a number is perfect or not.
4. A company insures its drivers if either of the following conditions are satisfied
    - Driver is married.
    - Driver is an unmarried, male and above 30 years of age.
    - Driver is unmarried, female and above 25 years of age.

   Write a program to decide if a driver is to be insured using logical operators.
5. Write a program to read a list of positive integers terminated by -1 and display the odd and even numbers separately and also their respective counts.
6. Write a program to read values of n and x and print the value of y using switch case where
    a. y=n+x when n=1
    b. y=1+x/n when n=2
    c. y= n+3x when n=3
    d. y=1+nx when n>3 or n<1.
7. Write a program to n values of sales and then calculate the commission on sales amount where the commission is calculated as follows:
    a. If sales <= Rs.500, commission is 5%.
    b. If sales > 500 but <= 2000, commission is Rs 35 plus 10% above Rs 500.
    c. If sales > 2000 but <= 5000,commission is Rs 185 plus 12% above Rs.2000.
    d. If sales > 5000 ,commission is 12.5%.
8. -
9. Write a program to find out minimum, maximum, sum and average of n numbers without using array.
10. Write a program to find out the roots of a quadratic equation. Use proper testing to find checks for real and complex roots.
11. Write a program to print the digits of a number in words. (eg. if a number 841 is entered through the keyboard your program should print “Eight Four One”.)

12. Write a function to return the HCF of two positive integers. Write a main function to read two positive integers and print their HCF and LCM by using the above function.
13. Write a program to convert a decimal number into binary number using function.
14. Write a program to display the result of sine series using function.
15. Write a program to find the sum of the following series

    1+x-x^3/3!+x^5/5!-x^7/7!+ . . . . . corrected up to the 3 decimal place.

16. Write functions to compute the factorial of a number using both recursive and non-recursive procedure.
17. Write a program to print the values of ${}^{n}c_{r}$ and ${}^{n}p_{r}$ for given positive integers $n \geq r > 0$. Use a function fact(n) to return the factorial of a non-negative integer.n.

    ${}^{n}c_{r}=n!/r!*(n-r)!$ ${}^{n}p_{r}=n!/(n-r)!$
18. Write a program to display the first n Fibonacci numbers using function.
19. Write a program to display the prime numbers within a given range. Write a function to check whether a given integer is prime or not and use it.
20. Write a program to Multiply two matrices using function
21. Write a function to check if a given square matrix is symmetric or not. Write a main function to implement it.
22. Write a program to merge two sorted arrays.
23. Write a program to implement selection sort using function.
24. Write a program to count the number of vowels in a string.
25. Write a program to concatenate two strings using function (without using library function).
26. Write a program to convert a string from upper case to lower case and vice versa.
27. Write a program to swap two numbers using function (pass the pointers).
28. Write a program using pointers to copy a string to another string variable (without using library function).

Part-II Data Structure

Total marks: 25 (Internal 5 and semester end examination 20)

Each student should do at least 10 assignments from the following list.

1. Implement binary search and linear search algorithms on arrays.
2. Implement following array sorting algorithms :
    i) Bubble sorting
    ii) Insertion sort
    iii) Selection sort
    iv) Quick sort
    v) Merge sort

3. Write a program to create a singly linked list and insert an element at the beginning, end, and at a given position of the linked list.

4. Write a program to create a singly linked list. Write functions for
   i. counting the number of elements in a list
   ii. to search for a given element in a list. If the item has been found then it should return the position at which the item was found; otherwise it should return -1 to indicate not found.

5. Write a function to concatenate two linked lists.

6. Write a function to merge two sorted linked lists.

7. Write a program to create a doubly linked list and insert an element at any position.

8. Write a program to create a doubly linked list and delete an element from a given position.

9. Write a program to create a circular linked list and display it.

10. Write a program to implement a stack using
    i) array structure
    ii) linked list structure

11. Write a program to evaluate a postfix expression using stack.

12. Write a program to implement a queue using array.

13. Write a program to implement a queue using linked list.

14. Write a program to create a binary search tree using link representation and display the elements in preorder, in order and post order using recursive function.

15. Write a program to create a binary search tree using link representation and
    i) search
    ii) delete an item from the binary search tree.

E401 INTRODUCTION TO DATABASE MANAGEMENT SYSTEM

Total Marks: 50 (Internal 10 and semester end examination 40)

UNIT-1 **marks(10)**

Database, Traditional file approach (File management system) vs Database management system. Characteristics of DBMS Advantages & disadvantages of DBMS, DBMS users, DBMS Architecture& data independance . Data model , Schemas & Instances .Three level architectural schema & Data -

independence. Relational model , RDBMS & its terminologies .Concept of keys (primary key, alternate key, candidate key, composite key , superkey and foreign key). Fundamental integrity constraints (entity integrity , domain integrity & referential integrity).

UNIT-2 **marks (15)**

Relational algebra (Set theoretic operations & select , project ,join (equi join, theta join , natural join, aggregate functions & grouping).

ANSI SQL -92 Standard: DDL, DML,SQL constructs (select ..from ... where ...Group by .. having ...order by), insert, delete, update. Definition & use of nested queries, constraints considers (NOT NULL, UNIQUE , CHECK , FOREIGN KEY).

UNIT-3 **marks (15)**

Database design :Phases of database design ,Entities & attributes , Entity type , Entity sets Relationship type, relationship sets & instances. Relationship degree , roles & recursive relationship. Constraints on relationship types (cardinality ratios for binary relationship, participation constraints). Dependency diagram and ER diagram, naming conventions & design issues .Conversion of ER diagram into relational table. Normalization: Informal design guidelines for relational schema, Functional dependencies, normal forms based on primary keys (1NF,2NF,3NF & BCNF). Functional dependency diagram and design of relational database from it.

Suggested Readings:

1. Fundamentals of data base management system, Elmasri & Navathe
2. Introduction to database management system, C.J. Date
3. An introduction to Database systems; Bipin C. Desai; Galgotia publications.
4. Database Systems - Concept, Design and Applications; S.K.Singh; Pearson Education.

E402 PRACTICAL

DBMS Total marks: 50 (Internal 10 and semester end examination 40)

Each student should do at least 5 assignments from the following list.

1. Create a table ***Employee*** with the following columns:
    Emp_no (numeric) primary key
    Emp_name (string)
    Join_date (Date)
    Basic_pay_fixed_at (numeric)
    Date_of_birth (Date)

Insert the following data into the table.

Emp_no.	Emp_name	Join_date	Basic_pay_fixed_at	Date_of_birth
1001	Charles Babbage	01-Jun-2000	8000.00	03-10-1973
1002	George Boole	01-Jul-2001	5000.00	04-12-1972
1003	E.F. Codd	01-Jun-2001	8000.00	06-03-1969
1004	Bill Gates	01-Jul-2003	5000.00	09-10-1995
1005	Tony Greig	01-Aug-2004	8000.00	04-05-1985

2. Create the following two tables and insert data into the tables.

Player (Roll no.→Primary Key)

Roll no.	Name
10	Vijay Amrithraj
20	Leander Paes
30	Mahesh Bhupathi
40	Sania Mirza

Match (Match_no→Primary key,Roll no→Foreign key)

Match_no	Roll_no.	Match_Date	Opponent
1	20	10-Jul-2008	Washington
2	30	12-Jan-2008	Sampras

3	20	12-Aug-2008	Borg
4	30	20-Mar-2008	Vijay

Perform the following two operations:

(i) Perform EQUIJOIN operation to retrieve data from both the files.

(ii) Perform OUTERJOIN operation to retrieve the unmatched records.

3. Design an ER diagram for a ***BANK*** database schema. To consider that each Bank can have multiple branches, and each branch can have multiple Accounts and Loans for customer. Also to specify the non weak & weak entity types, key attributes & key types, relationship types, instances, constraints and participations.

4. Create a table ***Student*** taking the attributes given bellow

Roll_no, Student_name, Address, Date_of_addmission, Class

Section and Contact_no.

Write appropriate queries to perform the following operations:

a) To insert values in the Student table.

b) To delete values from Student table

c) To list the names of all students which roll_no > 20.

d) To search for students who got admitted before 01-01-2006.

e) To change the name of the student whose roll number is 10 to Amar.

5. Create tables ***Department*** and ***Employee*** with the attributes given bellow.

Employee (EmpNo , Empname, Address, Dno)
Department (Dno, Dname, Location)

Dno in Employee is a foreign key.
Write appropriate queries to perform the following operations:

a) To insert values in the tables.

b) To retrieve the names and addresses of all Employees working in the Finance department.

c) To print the location where Administration department is located.

d) to delete all information regarding a particular employee.

6. Create table ***Student*** and ***Course*** taking the attributes given bellow.

Student (Roll_no, Name, Semester, Course_no(Foreign key))

Course (Course_no, Course_name)

Write appropriate queries for the following operations:

a) To retrieve names of all students who are admitted to the course ‘BCA’.

b) To retrieve the names of all students whose course name is ‘BCA’ and who are in the 3rd semester.

c) To display details of all courses in which more than 100 students got admitted.

d) For course being offered, display the course name and number of students admitted to the course.

7. Create tables ***Employee***, ***Department***, ***Location***, ***Works_on***, and ***Project*** taking the attributes given bellow.

Employee (Fname, Lname, Empno, Bdate, Address, Salary, Dnumber)

Department (Dname,Dnumber,Mgrno)

Locations(Dnumber, DLocation)

Works_on(Empno, Pnumber, Hours_per_day)

Project(Pname, Pnumber, Location,Dnumber (Foreign))

Dependent(Empno, Dependent_name, Sex, DOB, Relationship)

Write appropriate queries for the following operations:

a) Retrieve the names and addresses of all employees who work in the Finance department.

b) To retrieve the names of all employees who works on all the projects controlled by department number 6

c) For each department, print the name of the department and the name of the manager of the department.
d) Retrieve the location where the Administration Department is located.
e) For every project located in Mumbai list the project number, the controlling department and department manger’s name and address.
f) Find out how many employees are there in each department.

g) Find the total salary of all employees of the “Research” department, as well as the maximum, minimum and average salary in this department

h) Retrieve the name of all employees who have no dependent.

i) Alter the “Employee” table by deleting the coloumn Bdate.

j) Retrieve the Fname, Lname of all employees whose salary is higher than average salary.

k) For each department retrieve the department number, the number of employee in the department and their average salary.

l) Retrieve the name of all employees who have two or more dependent

m) Retrieve the details of all employees who works on project number 1,2,3

8. Create Table

Client_master (Client_no, name, address, Bdue)

Product_master(P_number,Description,saleprice,costprice)

Sales_master(Salesmno,Sname,Addres,Salamnt,Remarks)

Sales_order(O_no,Client_no,Odate,Delyaddr,Salesmno)

Sales_order_detail(Order_no,Product_no, Qtyorder, product_rate,Qty_dispatched)

Write appropriate queries to perform the following operations:

i) List name of all clients having ‘a’ as the second letter in their names.

ii) Retrieve the description and total Qty sold for each product.

iii) Find product no. and description of non moving products (i.e product not being sold).

iv) For each product being sold, list the product number and the total amount (in Rs.) sold.

v) List all client who stay in ‘Bangalore’ or ‘Mumbai’

vi) List the clients who stay in a city whose First letter is ‘M’

vii) Find the names of clients who had purchased the item ‘Trouser’.

viii) Find out if T-Shirt’ has been ordered by any client and if so print the details of the client.

ix) List details of all products whose unit price is more than Rs. 5000.00.

x) Calculate the total amount (in Rs.) purchased by each client that has purchased items amounting more than Rs. 20000.

E501 COMPUTER ORGANIZATION AND OPERATING SYSTEM

Full Marks: 100 (Internal 20 and semester end examination 80)

Part - I Computer Organization: 70 marks (Internal 14 and semester end examination 56)

Normalized floating point numbers: **Marks 7**

Representation and arithmetic operations with normalized floating point numbers.

Combinational circuit: **Marks 10**

Half-adder, Full-adder, decoder, encoder, Multiplexer, De-multiplexer, Magnitude Comparator including design of each.

Sequential circuit: **Marks 13**

Simple R-S flip-flop or Latch, D Flip-flop, Clocked R-S Flip-flop. J-K flip-flop, Master-Slave Flip-flop, J-K Master-Slave Flip-flop, Design of Counter Registers: Shift registers, Binary counter (synchronous and asynchronous).

Components of a computer: **Marks 14**

(8085A microprocessor should be considered)

Block diagram of computer showing CPU, Main memory, I/O and address, data and control bus and a brief description of each component. Block diagram of a CPU showing ALU, general purpose registers, special purpose registers like PC, SP, IR, PSW and condition flags, internal bus, control unit, timing unit, instruction decoder etc and a brief description of each component. Basic idea of memory read and memory write and also how instructions are fetched and executed. Machine cycle & Instruction Cycle.,

Instruction format. **Marks 12**

Addressing modes, Overview of 8085 Instruction set, Instruction Format, Opcode,effect of instructions on flags. Communication between CPU and main memory and CPU and I/O. Methods of I/O: -, Interrupt Driven I/O, Programmed I/O, Interrupt and DMA, DMA Transfer. I/O buses, device controller, Memory mapped I/O & I/O Mapped I/O.

Suggested Readings:

1. Computer organization; ISRD Group; Tata-McGraw Hill.
2. Computer Architecture and Organization – Design principles and Applications: B Govindarajalu; Tata-McGraw Hill

Part - II Operating System 30 Marks (Internal 6 marks and semester end examination 24)

No theory questions are to be set from the following two units. The theory classes will prepare the students for the practical assignments in paper E502.

Windows OS-An overview of different versions of windows, Basic windows elements. File management through windows. Widows accessories, windows Explorer, Entertainment system tools, Understanding OLE. Installation.

Linux OS : Introduction to Linux, Files and directories, architecture(kernel, shells, utilities) and various Linux commands. File manipulations, redirection and filters , editors(vi, ed etc.) Concept of process, System administration: File system, system administration commands.

Concepts of OS Memory management 8 marks

Memory management, Concept of Cache memory and virtual memory, paging and segmentation. Introduction to page replacement algorithms (NRU, FIFO, LRU)

Process management and introduction to deadlock 8 marks

Process and process states. Scheduling. Introduction to scheduling algorithms (First Come First Serve, Round Robin) . Concept of deadlock (only introduction).

Shell Programming 8 marks

Introduction- the Bourne Shell, basic shell programming. The shell- file security & permissions, running commands. Shell Programming examples – (simple arithmetic calculation, degree-fahrenheit-celsius calculation, string handling, condition testing, checking prime numbers).

Suggested Readings:

1. Sumitabha Das; UNIX concepts & applications; Tata McGraw Hill, New Delhi; (fourth Edition).
2. A.S. Tanenbaum; Moderrn Operating Systems; Prentice Hall of India, New Delhi, 1995; (Third Edition).
3. M. G. Venkateshmurthy; Introduction to Unix & Shell Programming; Pearson Education

E502 PRACTICAL

Operating System and ICT Hardware
Full Marks: 100 (Internal 20 and Semester end examination 80)

Objectives:

The Practical introduces the students to a personal computer and its basic peripherals, the process of assembling a personal computer, installation of system software like Windows OS, LINUX OS, device drivers. Basic system administration in Linux which includes: Basic Linux commands in bash, Create hard and symbolic links, Text processing, Using wildcards In addition hardware and software level troubleshooting process, tips and tricks would be covered.

Different ways of hooking the PC on to the network and internet from home and workplace and effectively usage of the Internet. . Usage of web browsers, email, newsgroups and discussion forums would be covered. In addition, awareness of cyber hygiene, i.e., protecting the personal computer from getting infected with the viruses, worms and other cyber attacks would be introduced.

Task 1: Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor.

Task 2: Every student should individually install MS windows on the personal computer. Lab instructor should verify the installation and follow it up with a Viva. Hard disk formatting and partitioning.

Task 3: Every student should install Linux on the computer. This computer should have windows installed. The system should be configured as dual boot with both windows and Linux. Lab instructors should verify the installation and follow it up with a Viva

Task 4: Learn basic commands in Linux

Task 5: Hardware Troubleshooting: Students have to be given a PC which does not boot due to improper assembly or defective peripherals. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed up with a Viva

Task 6: Software Troubleshooting: Students have to be given a malfunctioning CPU due to system software problems. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed up with a Viva.

Task 7: The test consists of various systems with Hardware / Software related troubles. Formatting disks without operating systems. Installation of antivirus software.

Task 8: Write shell programs as mentioned in paper E501.

M601 OBJECT ORIENTED PROGRAMMING IN C++ AND COMPUTER NETWORKS

Full marks: 100
Part I: Object Oriented Programming using C++

Marks: 60 (Internal 12 and Semester end examination 48)

Introduction: **Marks: 5**

What is OOP . Introducing Object-Oriented Approach, Relating to other paradigms. Benefits of OOP and methods.

Basic terms and ideas: **Marks: 10**

Abstraction, Encapsulation, Inheritance, Polymorphism, Review of C, Difference between C and C++ - cin, cout, new, delete operators. Functions in C++: main function, function prototyping, call by reference, return by reference, functions- inline, friend.

Classes and Objects: **Marks: 20**

Encapsulation, information hiding, abstract data types, Object & classes, attributes, functions, C++ class declaration, member functions, State identity and behavior of an object, friend functions, constant member functions. Constructors and destructor, instantiation of objects. Default parameter values.

Operator overloading: Overload unary, binary operators, overloading binary operators using friends, manipulation of strings using operators.

Inheritance: **Marks: 7**

Inheritance- single inheritance, derivation using public, private & protected mode.

Polymorphism: **Marks: 6**

Polymorphism, Categorization of polymorphism techniques: compile time polymorphism, Polymorphism by parameter, run time polymorphism- only the concept is to be introduced without going into implementation details.

Part - II Computer Networks

Total Marks: 40 (Internal 8 and Semester end examination 32)

Overview: The uses of computer networks, Network topologies. Layered Architecture, Relationship of service to protocol, Interface between layers, Connection oriented vs. connectionless service, The OSI reference model. TCP/IP Model. **Marks: 5**

Physical layer: Modulation and demodulation. Guided and wireless transmission media, satellite communication and their relative merits and demerits. **Marks: 5**

Data link layer: Functions of data link layer: Framing, error control, flow control, link management.

Marks: 5

Network layer: Virtual circuit vs. datagram subnet. Establishment of connection in connection oriented service, optimality principle, Routing Algorithms: distance vector routing.

Definition of Flow control ,congestion and quality of service **Marks: 6**

Transport Layer

Works of the transport layer, Basic functionality of transport layer. **Marks: 6**

The application Layer:

Electronic mail, Mail transfer Agent & Mail user agent, Ftp, Telnet, WWW, HTTP, HTML, URL, Architecture of the World Wide Web. **Marks: 5**

Suggested Readings:

1. Computer Networks By A. Taneunbaum
2. Data Communication and Computer Networks by W. Stallings

E602 Practical

Total Marks: 100

Part I Object Oriented Programming using C++

Total Marks: 70 (Internal 14 and Semester end examination 56)

Each student should do at least 10 of the following assignments.

1. Define a class named ***triangle*** to represent a triangle using the lengths of the three sides. Write a constructor to initialize objects of this class, given the lengths of the sides. Write member functions to check
   (a) if a triangle is isosceles
   (b) if a triangle is equilateral.

Write a main function to test your functions.

2. Define a structure **employee** with the following specifications.

   Empno: Integer
   Ename: 20 character
   Basic, hra, da: float

   Calculate() : a function to compute net pay as basic+hra+da with float return type.
   Getdata(): a function to read values for empno, ename, basic, hra, da.
   Dispdata(): a function to display all the data on the screen

Write a main program to test the program.

3. Define a class ***circle*** to represent circles. Add a data member ***radius*** to store the radius of a circle. Write member functions area() and perimeter() to compute the area and perimeter of a circle.

4. Define a class ***complex*** with two data members ***real*** and ***imag*** to represent real and imaginary part of a complex number.

   Write member functions

   rpart(): to return the real part of a complex number

   ipart() : to return the imaginary part of a complex number

   Add() : to add two complex numbers.

   Mul() : to multiply two complex numbers.

   Write constructors with zero, one and two arguments to initialize objects.

   // This is an example of static polymorphism.

5. Define a class ***point*** with two data members ***xordinate*** and ***yordinate*** to represent all points in the two dimensional plane by storing their x co-ordinate and y co-ordinate values.

   Write member functions

   dist(): to return the distance of the point from the origin.

   slope(): to return the slope of the line obtained by joining this point with the origin.

   Write constructors with zero, one and two arguments to initialize objects.

   Write a friend function to compute the distance between two points.

6. Define a class ***String*** with the following data members
   char *p;
   int size
   and write member functions to do the following (without using library function) and using dynamic memory allocation.
   - Length of the string
   - Compare two string
   - Copy one string to another.
   - Reverse the string.
   - Write suitable constructors and destructors. Write a copy constructor for the class.

7. For the class ***complex*** defined in 3 above, overload the <<, >>, + and * operators in the usual sense. .
8. For the class ***string*** defined in 5 above, overload the <<, >> and + operators where + is to be used for concatenating two strings.

9. Define a class to store matrices. Write suitable friend functions to add and multiply two matrices

10. Define a class ***student*** with the following specification:

rollno	:integer
sname	:20 characters

Derive two classes ***artsst*** and ***scst.*** The class astst will represent students belonging to Arts stream and the class scst will represent students belonging to science stream. The artsst class will have additional data members ***ph, hs, en*** and **as** to store marks obtained by a student in three subjects Philosophy, History, English and Assamese. The class scst will have additional data members ***ph, ch, ma and eg*** to store marks obtained in Physics, Chemistry, Mathematics and English.

Write the following member functions in the classes artst and scst

ctotal() : A function to calculate the total marks obtained by a student takedata() : function to accept values of the data members

Showdata() : function to display the mark sheet of a student .

Part II Computer Networks

Total Marks: 30 (Internal 16 and Semester end examination 24)

Each student should do all of the following assignments.

1. Configure a LAN in Linux environment. Configure the NIC using ifconfig command. Also explore the following commands

   a. ifup
   b. ifdown
   c. ping

2. Learn the significance and usage of /etc/hosts file

3. Starting and stopping network services in Linux system.

4. Exercising remote login using **ssh** (The secure shell) and remote copy using **scp** (secure copy)

5. Write a connection oriented TCP/IP socket program in C where a client will request for a file to the server and the server will send the file to client. The IP address of the server and file name will be given as command line argument (Use loopback interface).

NOTE:

The students should be introduced with LAN, sockets, port numbers as well as Linux operating system during the theory sessions.