

ASSIGNMENT

2017 - 18



Computer Science(083)
CLASS - XII

DAV PUBLIC SCHOOL, MCL
KALINGA AREA, TALCHER, ANGUL - 759148

UNIT-WISE MARKS DISTRIBUTION

Duration: 3 hours

Total Marks: 70

Unit No.	Unit Name	Marks
1.	OBJECT ORIENTED PROGRAMMING IN C++	30
2.	DATA STRUCTURE	14
3.	DATABASE MANAGEMENT SYSTEM AND SQL	8
4.	BOOLEAN ALGEBRA	8
5.	COMMUNICATION TECHNOLOGIES	10
Total		70

Unit 1: Object Oriented Programming in C++ (50 Theory + 40 Practical) Periods

REVIEW: C++ covered In Class – XI,

Object Oriented Programming: Concept of Object Oriented Programming – Data hiding, Data encapsulation, Class and Object, Abstract class and Concrete class, Polymorphism (Implementation of polymorphism using Function overloading as an example in C++); Inheritance, Advantages of Object Oriented Programming over earlier programming methodologies,

Implementation of Object Oriented Programming concepts in C++: Definition of a class, Member of a class – Data Members and Member Functions (methods), Using Private and Public visibility modes, default visibility mode (private); Member function definition: inside class definition and outside class definition using scope resolution operator (::); accessing members from object (s), Objects as function arguments—pass by value and pass by reference;

Constructor and Destructor: Constructor: special characteristics, declaration and definition of a constructor, default constructor, overloaded constructors, copy constructor, constructor with default arguments;

Destructor: Special Characteristics, declaration and definition of destructor;

Inheritance (Extending Classes): Concept of Inheritances, Base Class, Derived classes, protected visibility mode; Single level inheritance, Multilevel inheritance and Multiple inheritance, Privately derived, publicly derived and Protectedly derived class, accessibility of members from objects and within derived class (es);

Data File Handling: Need for a data file, Types of data files – Text file and Binary file; **Text File: Basic** file operations on text file: Creating/Writing text into file, Reading and Manipulation of text from an already existing text File (accessing sequentially).

Binary File: Creation of file, Writing data into file, Searching for required data from file, Appending data to a file, Insertion of data in sorted file, Deletion of data from file, Modification of data in a file; Implementation of above mentioned data file handling in C++; Components of C++ to be used with file handling: Header file: fstream.h; ifstream, ofstream, classes; Opening a text file in—in, out, and app modes;

Using cascading operators (>>,<<) for writing text to the file and reading text from the file; open(), get (), read (), put (), write(), getline() and close() functions; Detecting end-of-file (with or without using eof() function), tellg(), tellp(), seekg(), seekp());

Pointers:

Introduction to Pointer, Declaration and Initialization of Pointer; Dynamic memory allocation/deallocation operators: **new, delete**; Pointers and Arrays: Array of Pointers, Pointer to an array (1 dimensional array), Function returning a pointer, Reference variables and use of alias; Function call by reference. Pointer to structure: De-reference/Deference operator: *, ->; self referential structure;

Unit 2: Data Structures (42 Theory + 36 Practical) Periods

Introduction to data structure- array, stack queues primitive and non-primitive data structure, linear and non-linear structure, static and dynamic data structure.

Arrays:

One and two Dimensional arrays: Sequential allocation and address calculation;

One dimensional array: Traversal, Searching (Linear, Binary Search), Insertion of an element in an array, deletion of an element from an array, Sorting (Insertion, Selection, Bubble)
Two-dimensional arrays: Traversal Finding sum/difference of two NxM arrays containing numeric values, interchanging Row and Column elements in a two dimensional array;

Stack (Array and Linked implementation of Stack):

Introduction to stack (LIFO: Last in First out Operations)
Operations on stack (PUSH and POP) and its Implementation in C++, Converting expressions from INFIX to POSTFIX notation and evaluation of Postfix expression;

Queue: (Array and Linked Implementation)

Introduction to Queue (FIFO: First in First out operations)
Operations on Queue (Insert and Delete and its Implementation in C++, circular queue using array).

Unit 3: Databases and SQL (20 Theory + 20 Practical) Periods

Data base Concepts: Introduction to data base concepts and its need.

Relational data model: Concept of domain, tuple, relation, key, primary key, alternate key, candidate key;

Relational algebra : Selection, Projection, Union and Cartesian product;

Structured Query Language:

General Concepts: Advantages of using SQL, Data Definition Language and Data Manipulation Language;

Data Types: NUMBER/DECIMAL, CHARACTER/VARCHAR/VARCHAR2, DATE;

SQL COMMANDS: CREATE TABLE, DROP TABLE, ALTER TABLE, UPDATESET....., INSERT, DELETE; SELECT, DISTINCT, FROM, WHERE, IN, BETWEEN, GROUP BY, HAVING, ORDER BY;

SQL functions: SUM (), AVG (), COUNT (), MAX () AND MIN (); Obtaining results (SELECT query) from 2 tables using equi-join, Cartesian product and Union

Note: Implementation of the above mentioned commands could be done on any SQL supported software on one or two tables.

Unit 4: Boolean Algebra (16 Theory + 0 Practical) Periods

Role of Logical Operations in Computing.

Binary-valued Quantities, Boolean Variable, Boolean Constant and Boolean Operators: AND, OR, NOT; Truth Tables; Closure Property, Commutative Law, Associative Law, Identity law, Inverse Law, Principle of Duality, Idempotent Law, Distributive Law, Absorption Law, Involution Law, DeMorgan's Law and their applications; Obtaining Sum of Product (SOP) and Product of Sum (POS) form the Truth Table, Reducing Boolean Expression (SOP and POS) to its minimal form, Use of Karnaugh Map for minimization of Boolean expressions (up to 4 variables);

Application of Boolean Logic: Digital electronic circuit design using basic Logic Gates (NOT, AND, OR, NAND, NOR)

Use of Boolean operators (NOT, AND, OR) in SQL SELECT statements, Use of Boolean operators (AND, OR) in search engine queries.

Unit 5: Communication Technologies (16 Theory + 3 Practical) Periods

Evolution of Networking: ARPANET, Internet, Interspace Different ways of sending data across the network with reference to switching techniques (Circuit and Packet switching).

Data Communication terminologies: Concept of Channel, Bandwidth (Hz, KHz, MHz) and Data transfer rate (bps, Kbps, Mbps, Gbps, Tbps).

Transmission media: Twisted pair cable, coaxial cable, optical fiber, infrared, radio link, microwave link and satellite link.

Network devices: Modem, RJ45 connector, Ethernet Card, Router, Switch, Gateway, wifi card.

Network Topologies and types: Bus, Star, Tree, PAN, LAN, WAN, MAN.

Network Protocol: TCP/IP, File Transfer Protocol (FTP), PPP, SMTP, POP3 Remote Login (Talent), and Internet Wireless/Mobile Communication protocol such as GSM, CDMA, GPRS, and WLL.

Mobile Telecommunication Technologies : 1G, 2G, 3G and 4G; Mobile processors; Electronic mail protocols such as SMTP, POP3, Protocols for Chat and Video Conferencing VOIP, Wireless technologies such as Wi-Fi and WiMax

Network Security Concepts:

Threats and prevention from Viruses, Worms, Trojan horse, Spams, Use of Cookies, Protection using Firewall, https; India IT Act, Cyber Law, Cyber Crimes, IPR issues, hacking.

Introduction To Web services: WWW, Hyper Text Markup Language (HTML), Extensible Markup Language (XML); Hyper Text Transfer Protocol (HTTP); Domain Names; URL; Website, Web browser, Web Servers; Web Hosting, Web Scripting – Client side (VB Script, Java Script, PHP) and Server side (ASP, JSP, PHP), Web 2.0 (for social Networking)

E-commerce payment transactions using online banking, mobile banking and payment apps and services.

Class XII (Practical) - C++

Duration: 3 Hours

Total Marks : 30

1. Programming in C++

10

One programming problem in C++ to be developed and tested in Computer during the examination.

Marks are allotted on the basis of following:

Logic : 6 Marks

Documentation/Indentation : 2 Marks

Output presentation : 2 Marks

Notes: The types of problem to be given will be of application type from the following topics .

Arrays (One dimensional and two dimensional)

Class(es) and objects

Stack using arrays and or linked implementation

Queue using arrays (circular) and or linked implementation

Binary File operations (Creation, Displaying, Searching and modification)

Text File operations (Creation, Displaying and modification)

2. SQL Commands

05

Five Query questions based on a particular Table / Relation to be tested practically on Computer during the examination. The command along with the result must be written in the answer sheet.

3. Project Work

05

The project has to be developed in C++ language with Object Oriented Technology and also should have use of Data files. (The project is required to be developed in a group of 2-4 students)

- Presentation on the computer
- Project report (Listing, Sample Outputs, Documentations)
- Viva

* 1 mark is for innovation while writing program.

4. Practical File

06

Must have minimum 20 programs from the following topics

- Arrays (One dimensional and two dimensional, sorting, searching, merging, deletion & insertion of elements)
- Class(es) and objects
- Stacks using arrays and linked implementation
- Queue using arrays & linked implementation (circular also).
- File (Binary and Text) operations (Creation, Updation, Query)
- Any computational Based problems
- 15 SQL commands along with the output based on any table/relation:

5. Viva Voce

04

Viva will be asked from syllabus covered in class XII and the project developed by student.

Assignment (2017-18)

CHAPTER 1, 2, 3 : C++ REVISION TOUR, OOP & FUNCTION OVERLOADING

- (a) Give the difference between the type casting and automatic type conversion. Also give a suitable C++ code to illustrate both.
(b) Difference between the formal and actual parameter. Also give a suitable C++ code to illustrate both.
(c) Differentiate between post increment and pre increment operators. Also give a suitable C++ code to illustrate both.
(d) What is the difference between Actual parameter and Formal parameters? . Also give a suitable C++ code to illustrate both.
(e) When a function is overloaded, there are multiple definitions of the functions. What make the various definition different from each other.

2. Find the output of the following program:

```
void convert(char msg[ ], char ch)
{
    for (int c =0; msg[c]!='\0' ; c++)
    { if ((msg[c] == 'R') || (msg[c] == 'r'))
      msg[c] -- ;
      else
      if ((msg[c] >= 'B') && (msg[c] <= 'W'))
      msg[c]= tolower(msg[c]);
      else
      if (isdigit(msg[c+1]))
      msg[c] = msg[c] + 1;
      else
      msg[c] = ch;
    }
}
void main ()
{
    char text [ ] = "PreBOARD2013";
    convert(text, '*');
    cout<<"NewText : "<<text<<endl;
}
}
```

3. Which C++ header file(s) will be essentially required to execute the following C++ code:

```
void main()
{
    int x=8; float res;
    res=sqrt(x);
    cout<<setprecision(2)<<res;
}
}
```

4. Find the output of the following program fragment.

```
void main()
{
    int A=5,B=5;
    for(int i=1;i<=2;i++)
    {
        cout<<"Line "<< i<<" = "<<A - -<<"&"<<B-2<<endl;
        cout<<"Line "<< i<<" = "<<++B<<"&"<<A+2<<endl;
    }
}
}
```

5. How many times the following loop will execute.

- int A=1;
while(++A<=5)
cout<<A++;
- for (i=0 ; i=5 ; i++);

6. Observe the following program, find the correct possible output(s) from the options, and justify your option.

```
void main( )
{
    randomize( );
    int x=10;
    for( int y = 1; y <= 4 ;y++)
        cout<<random(x - -)+y<<" ";
}
```

Output Option:

- (i) 10 : 12 : 11 : 10 : (ii) 12 : 10 : 11 : 12 :
 (iii) 9 : 9 : 9 : 9 : (iv) 11 : 12 : 9 : 8 :

7. In the following program, find the correct possible output(s) from the options:

```
void main ( )
{
    char serial[ ] = {'A', 'B', 'C', 'D'};
    int number[ ] = { 2, 6, 7, 3};
    randomize();
    cout << " The winner is : ";
    cout << serial [random(3)];
    for (int i = 0; i < 4; i++)
        cout << number[sizeof(int) + random(2) - 1 ];
}
```

Outputs:

- (i) The winner is : A2776 (ii) The winner is : D6766
 (iii) The winner is : B6767 (iv) The winner is : C3672

8. Find the output of the following program segment.

```
Struct land
{
    int length;
    int breadth;
    float area;
};
void calarea(land &p1, int y=5)
{
    p1.area=p1.length*p1.breadth;
    p1.area /=y;
    p1.length++;
    p1.breadth++;
}
void main( )
{
    land first = { 10, 12}, second = {5, 10};
    calarea(first);
    cout<<first.area<<"#"<<first.length<<"#"<<first.breadth<<endl;
    calarea(second,2);
    cout<<second.area<<"#"<< second.length<<"#"<< second.breadth<<endl;
}
```

9.(i) Rewrite the following program after removing the syntactical errors. Underline each correction.

```
type def int integer;
struct number
{
    integer a[5]; }
void main( )
{
    number x;
    for(int i=0; i<5, i++)
        cin>>x[i].a;
}
```

(ii) What will be the output of the following segment.

```
#include <iostream.h>
enum color{Red, green, blue, yellow};
void main( )
{
    cout<<blue; cout<<green;
}
```

(iii) What will be the output of the following segment?

```
void main()
{
    int a[5] = {3, 4, 5, 6, 7};
    int i, j, k = 1, m;
    i = ++a[1];
    j = a[2]++;
    m = a[i++ - 3];
    cout<<i<<j<<m;
}
```

10. What will be the output of following program.

```
int sum(int a)
{
    return a+a;
}
int sum(int a,int b)
{
    return a+b;
}
float sum(int a, float b)
{
    return (a+b);
}
void main( )
{
    int n1=2,n2=4;
    float n3=5.3;
    cout<<sum(n1)<<endl;
    cout<<sum(n2,n3)<<endl;
    cout<<sum(n1,n2)<<endl;
}
```

11. What will be the output of following program segment.

```
int max(int x, int y)
{
    return(x>y ? x : y); }
int max(int x, int y, int z)
{
    int m = (x>y ? x : y);
    return((z>m) ? z : m); }
void main( )
{
    cout<<max(99,77)<<endl;
    cout<<max(32, 41, 65)<<endl;
}
```

CHAPTER 4 : CLASSES AND OBJECTS

1. What do you understand by data encapsulation and data hiding? Also give a suitable C++ code to illustrate both?

2. What are static class members?

3. How is working of non member function different from a friend function ?

4. What is an inline function ? What are its advantages and disadvantages.

5. Define a class **Book** with following specifications :

Private members of the class Book are

Book_no integer

Book_title 20 character

Price float

Total_cost() A function to calculate the total cost for N number of copies, where N is passed to the function as argument.

Public members of the class Book are

Input() Function to read Book_no, Book_title and price

Purchase() Function to ask the user to input the number of copies to be purchased. It invokes Total_cost() and prints the total cost to be paid by the user.

6. Define a class **RESORT** with the following specification :

Private members:

rno to store room no between 101 and 909

name to store customer name

charges to store per day charges

days to store number of days of stay

compute() a function to calculate and return amount as days * charges and if the value of days*charges is more than 11000 then as 1.02 * days * charges

Public members of class worker

get_info() a function to accept values for rno, name, charges, days.

disp_info() a function to display rno, name, charges, days and amount(amount to be displayed by calling function compute()

7. Define a class **power** with the following specifications :

private members

mno(meter no) long

cname(cunsumer's name) 20 character

units(unit consumed) float

charges(charges to be paid) float

calccharges() to calculate charges according to the following conditions

<u>units</u>	<u>charges</u>
below 100	Rs2.00 per unit
100 – 200	Rs3.00 per unit
more than 200	Rs5.00 per unit

public members

enterdata() function to accept values for mno, cname, units and call the function calccharges() to calculate the charges.

showdata() function to display all the data members on the screen

Write the main function to manage the power for 10 consumers.

8. Define a class **clock** with following specifications:

Private Members

Hour of type integer

Minute of type integer

Public members

Readtime(int h, int m) to initialize the data members

Addtime(clock t1, clock t2) to find the sum of two time(assuming 12 hour clock time)

Showtime() to display the final time

Write a program to input two different objects and display their sum.

9. Define a class named **admission** in C++ with the following descriptions:

Private Members:

Ad_no integer(Ranges 10 – 2000)
Name string
Class string
Fees Float

Public Members:

Function Read_data() to read an object of admission type

Function Display() to display the details of an object

Function Draw_nos() to choose 2 students randomly and display the details. Use random function to generate admission nos. to match with ad_no.

10. What will be the output of the following program segment :

```
class myclass
{
int a;
static int b;
public :
void set(int num);
void print( );
};
void myclass :: set(int num)
{
a = num;
b++;
}
void myclass :: print( )
{
cout<<"a="<<a<<endl;
cout<<"b="<<b<<endl;
}
int myclass::b = 0;
void main( )
{
myclass ob1, ob2;
ob1.set(10);
ob1.print( );
ob2.set(20);
ob2.print( );
}
```

11. Rewrite the following C++ program code after removing the syntax errors. Underline each corrections.

```
class train
{
long tno=18012;
char description[45];
public
void entry( ) { cin>>tno; gets(description); }
void display( ) { cout<<tno<<" : "<<description<<endl; }
}
void main()
{
train t;
entry.t( );
t.show( );
}
```

CHAPTER 5 : CONSTRUCTOR & DESTRUCTOR

1. What do you understand by constructor & Destructor functions used in classes ? How are these functions different from other functions.
2. Can there be multiple constructors of the same class ?
3. What is a copy constructor ? What do you understand by constructor overloading ?
4. Given the following C++ code, answer the questions (i) to (iv)

```
class molecule
{
    char mol_name[20];
    int mol_size;
    public :
    molecule( )           // function 1
    { strcpy(mol_name,"carbon");
      mol_size=4;
    }
    molecule(char *p, int n)   // function 2
    {
      strcpy(mol_name,p);
      mol_size=n;
    }
    molecule(molecule &m);    // function 3
    ~molecule( );           // function 4
};
```

- (i) Which feature of Object oriented programming is implemented by the functions function1, function2, function3 and function4.
 - (ii) Write complete definition for function3
 - (iii) What is the scope of the data members of the class molecule? What does the scope of the data members depend on.
 - (iv) Create objects such that it invokes function1 and function2 respectively.
5. What will be the output of the following code.

```
class A{ public:
    A( )
    { cout<<"Constructor A"<<endl; }
    ~A( )
    { cout<<"Destructor A"<<endl; }
};
class B{ public:
    B( )
    { cout<<"Constructor B"<<endl; }
    ~B( )
    { cout<<"Destructor B"<<endl; }
};
class C{ A oa1;
        B ob1;
    public:
    C( )
    { cout<<"Constructor C"<<endl; }
    ~C( )
    { cout<<"Destructor C"<<endl; } };
void main( )
{
    C oc1;
    B ob2;
    A oa2, oa3;
}
```

6. What will be the output of the following program code.

```
class date
{
    int mo, da, yr;
    public:
    date()
    { mo = 0; da = 0; yr = 0; }
    date(int m, int d, int y)
    { mo = m; da = d; yr = y;}
    ~date();
    void display()
    { cout<<"\n"<<mo<<"/"<<da<<"/"<<yr; }
};
date::~date()
{ cout<<"\n Date Destructor " ;}
void main()
{
    date d[2];
    date temp(6, 4, 10);
    d[0] = temp;
    d[0].display();
    d[1].display();
}
```

7. Give the output of the following program.

```
class myclass
{
    int a;
    public:
    myclass()
    {
        cout<<"In Constructor\n";
        a = 5;
    }
    ~myclass() { cout<<"Destructor Out \n"; }
    void show() { cout<<a<<"\n"; }
};
void main()
{
    myclass ob;
    ob.show()
}
```

8. What will be the output of the following program ?

```
class metro
{
    int mno, tripno, count;
    public:
    metro(int tmno=1) { mno=tmno; tripno=0; count=0; }
    void trip(int pc=20) { tripno ++; count + = pc; }
    void showstatus() { cout<<mno<<":"<<":"<< tripno<<":"<<count<<endl; }
};
void main()
{
    metro m(10),t;
    m.trip();
    t.trip(50);
    m.showstatus();
    t.showstatus();
}
```

9. Define a class **Clothing** in C++ with the following descriptions:

Private members:

Code of type string
Type of type string
Size of type integer
Material of type string
Price of type float

A function **calprice()** which calculates and assigns the value of price as follows:

For the value of material as "COTTON":

<u>Type</u>	<u>Price(Rs)</u>
TROUSER	1500
SHIRT	1200

For material other than "COTTON" the above mentioned price gets reduced by 25%

Public members :

A **constructor** to assign initial values of code, type and material with the word "NOT ASSIGNED" and size and price with 0

A function **enter()** to input the values of the data members code, type, size and material and invoke the calprice() function

A function **show()** which displays the content of all the data members for a clothing

10. Define a class **Play** in C++ with the following specifications :

Private members of class play

Playcode integer
Playtitle 25 character
Duration float
Noofscenes integer

Public member function of class play

A **constructor** function to initialize duration as 45 and noofscenes as 5

Newplay() function to accept values for playcode and playtitle

Moreinfo() function to assign the values of duration and noofscenes with the help of corresponding values passed as parameters to this function

Showplay() function to display all the data members on the screen

11. Define a class **competition** in C++ with the following specifications :

Private members

event_no integer
description 25 character
score integer
qualified char

Public members

A **constructor** function to initialize event_no as 1001, description as "state level", score as 50 and qualified as 'n'

input() function to take the input for event_no, description and score

award(int) to award qualified as 'y' , if score is more than cutoffscore(passed as argument) else 'n'

show() function to display all the data members on the screen

CHAPTER 6 : INHERITANCE – EXTENDING CLASSES

1. How does inheritance influence the working of constructor and destructor?
2. What is containership ? How does it different from inheritance ?
3. Consider the following and answer the questions given below.

```
class Goods
{ int id;
protected :
char name[20];
long qty;
void Incr(int n);
public :
Goods();
~Goods();
void get(); };
class Food_products : public Goods
{ char exp_dt[10];
protected :
int id;
int qty;
public :
void getd();
void showd(); };
class Cosmetics : private Goods
{ int qty;
char exp_date[10];
protected :
int id;
public :
~Cosmetics();
Cosmetics();
void show();
};
```

- (i) How many bytes will be required by an object of class Food_products.
 - (ii) Name the member functions accessible through the object of class Food_products.
 - (iii) From the following, Identify the member function(s) that cannot be called directly from the object of class Cosmetics show(), getd(), get()
 - (iv) If the class cosmetics inherits the properties of food_products class also, then name the type of inheritance.
4. Consider the following and answer the questions given below.

```
class Mydata
{ protected:
int data;
public:
void Get_mydata(int);
void Manip_mydata(int);
void Show_mydata(int);
Mydata( );
~Mydata( );
};
class Personal_data
{ protected:
int data1;
```

```

public:
void Get_personaldata(int);
void Show_personaldata(int);
Personal_data1 ( );
~Personal_data1 ( );
};
class Person: public Mydata, protected Personal_data
{ public:
void Show_person(void);
Person ( );
~Person ( );
};

```

- i) How many bytes will be required by an object belonging to class Person?
- ii) Which type of inheritance is depicted in the above example?
- iii) List the data members that can be accessed by the member function Show_person().
- iv) What is the order of constructor execution at the time of creating an object of class Person?

5. Answer the questions (i) to (iv) based on the following:

```

Class ONE
{ int one_no;
char one_name[20];
protected:
void register();
public:
ONE();
void status();
};
class TWO
{ int two_no;
char two_name[20];
protected:
float salary;
public:
TWO();
void enter();
void show();
};
class THREE :private ONE, public TWO
{ char three_no[10];
char Three_date[8];
public:
THREE();
void sales_entry();
void sales_detail();
};

```

- (i) Write the names of data members which are accessible from objects belonging to class ONE.
- (ii) Write the names of all the members which are accessible from objects belonging to TWO
- (iii) Write the names of all the members which are accessible from member functions of class THREE
- (iv) How many bytes will be required by an object to class TWO.

6. Consider the following class declarations and answer the questions.

```

class bank
{ protected :
int capital;

```

```

public :
void get_data( );
void display( ); };
class loans : public bank
{protected :
float house_L;
float car_L;
public :
float calc( );
void show( );
};
class funds: protected loans
{ protected :
float value;
public :
float interest(int); };

```

- (i) List the names of data and functions inherited by class funds.
- (ii) Declare an array of objects with 30 elements of class loans.
- (iii) Is the function get_data() of class bank accessible to objects of class loans?
- (iv) Is data member value of funds accessible to objects of class funds ?

7. Answer the questions (i) to (iv) based on the following code:

```

class chairperson
{ long cid;
char cname[20];
protected:
char description[40];
void allocate();
public:
chairperson();
void assign();
void show();
};
class director
{ int did;
char dname[20];
protected:
char profile[30];
public:
director();
void input();
void output();
};
class company : private chairperson , public director
{ long id;
char city[10],country[15];
public:
company();
void enter();
void display();
};

```

- (i) Write the names of data members, which are accessible from objects belonging to class company.
- (ii) Write the name of all the members which are accessible from member functions of class company.
- (iii) Which type of inheritance is illustrated in the above C++ code.
- (iv) How many bytes will be required by an object belonging to class director?

CHAPTER 7 : DATA FILE HANDLING

1. How are binary files different from text files in C++ ?
2. What is a stream ? Name the streams generally used for file I/O ?
3. Write two member functions belonging to fstream class.
4. What is the difference between the functioning of **ios::ate** and **ios::app** file modes ?
5. Observe the program segment given below carefully and answer the question that follows

```
class school
{ private :
  char name[25];
  int numstu;
public:
  void inschool( );
  void outschool( );
  int retnumstu( ) { return numstu; }
};
void modify(school A)
{ fstream INOUT;
  INOUT.open("school.dat",ios::binary|ios::in|ios::out);
  school B;
  int reread=0, found=0;
  while(!found && INOUT.read((char*)&B,sizeof(B))
  { reread++;
  if(A.retnumstu( ) == B.retnumstu( ))
  {
  _____//missing statement1
  _____//missing statement2
  Found=1;
  }
  else
  INOUT.write((char*)&B,sizeof(B));
  }
  if(!found)
  cout<<"\nRecord for modification does not exist";
  INOUT.close( );
  }
}
```

If the function **modify()** is supposed to modify a record in file **school.dat** with the values of school A passed to its argument. Write the appropriate statement for statement1 using **seekp()** or **seekg()** to position the file write pointer to an appropriate place in the file and statement2 to perform the write operation.

6. Write a function **RevText()** to read a text file "Input.txt" and Print each word in reverser order

Example :

If value in text file is : INDIA IS MY COUNTRY

Output will be: AIDNI SI YM YRTNUOC

7. Write a function in C++ to count the word "this" and "these" present in a text file "article.txt".
8. Write a function in C++ to read and display the details of all users whose status is 'A'(i.e. Active) from a binary file "user.dat". Assuming the binary file "user.dat" is containing objects of class user, which is defined as follows:

```
class user
{ int uid;           //user id
  char uname[25];   //user name
  char status;      //user type : A active | Inactive
public:
  void register( )  { cin>>uid; gets(uname); cin>>status;}
  void show( )      { cout<<uid<<uname<<status; }
  char getstatus( ) { return status; }
};
```

9. Consider the following class declaration.

```
class mobile
{
    char number[10];
    int calls;
    public:
    void enter()          { gets(number); cin>>calls; }
    void billing()       { cout<<number<<"# "<<calls<<endl; }
    int getcalls( )      { return calls; }
};
```

Give function definitions to the following.

(i) Write a function in C++ to accept the objects of class mobile from the user and write to a binary file "mobile.dat".

(ii) Write a function in C++ to read the objects of class mobile and display all the objects on the screen which have more than 1000 calls from binary file "mobile.dat".

10. Declare a structure in C++ telerec, containing name(20 characters) and telephone number. A binary data file "TELE.DAT" stores data of the type telerec. Write a program to display the name for a given telephone number. If the telephone number doesnot exist then displays error message "Record not found."

11. Following is the structure of each record in a data file named "vehicle.dat".

```
struct vehicle
{
    int vcode;
    char vname[20];
    float cost;
};
```

Write a function in C++ to update the file with a new value of cost for a particular vehicle. The value of vcode and cost are read during the execution of the program.

12. Assuming the class Computer as follows:

```
Class computer
{ char chiptype[10];
  int speed;
  public:
  void getdetails( ) { gets(chiptype); cin>>speed; }
  void showdetails( ) { cout<<"Chip"<<chiptype<<"Speed="<<speed<<endl; }
};
```

Write a function readfile() to read all the records present in an already existing binary file "**CHIP.DAT**" and display them on the screen, also count the number of records present in the file.

13. Write a function in C++ to read the content of a text file "PLACES.TXT" and display all the lines on screen, which are either starting with 'P' or 'S'.

CHAPTER 8 : POINTERS

1. Write the difference between static and dynamic variable.
2. What do you mean by self referential structure ?
3. Distinguish between `int *ptr = new int(5)` ; and `int *ptr = new int[5]`;
4. Give the output of following program.

```
void main()
{
    char *notes;
    char str[ ]="VERYgood";
    notes=str;
    while(*notes!='\0')
    {
        *notes=(isupper(*notes)?tolower(*notes):toupper(*notes));
        cout<<notes<<endl;
        notes++;
    }
}
```

5. Give the output of the following program.

```
void main()
{
    int array[ ] = {10, 20, 30, 40};
    int *arptr = array;
    int value = *arptr;    cout<<value<<'\n';
    value = *arptr++;    cout<<value<<'\n';
    value = *arptr;    cout<<value<<'\n';
    value = *++arptr;    cout<<value<<'\n';
}
}
```

6. Predict the output of the following program.

```
void main()
{
    int a[5] = {13, 15, 16, 18, 19};
    change(a);
    for(int i = 4; i>=0; i- -)
        cout<<a[i];
    cout<<"\n";
}
void change(int *b)
{
    for(int i=0; i <= 4 ; i++)
        { if(*b%3 != 0)
            *b = *b +1;
          b++;
        }
}
}
```

7. What will be the output of the following program.

```
void main()
{
    char *a[2]={"Amit","Sumit"};
    for(int i=0;i<2;i++)
    { int l=strlen(a[i]);
      for(int j=0;j<l;j++,a[i]++)
          cout<<*a[i]<<":";
      cout<<endl;
    }
}
```

CHAPTER 9 : ARRAYS

1. Assume an array E containing elements of structure Employee is required to be arranged in descending order of Salary. Write a C++ function to arrange the same with the help of bubble sort, the array and its size is required to be passed as parameters to the function. Definition of structure Employee is as follows:

```
struct Employee
{
    int Eno;
    char Name [25];
    float Salary;
};
```

2. Write a C++ function revdup(int x[], int n) to remove the duplicate occurrence of the value present in an integer array.

For e.g.If array initially x[] = { 1,1,1,7,2,2,6 }

After removing duplicate values the array will be x[] = { 1,7,2,6 }

3. Given two arrays of integers X and Y of sizes m and n respectively. Write a function named MERGE() which will produce a third array named Z, such that the following sequence is followed.

(i) All odd numbers of X from left to right are copied into Z from left to right.

(ii) All even numbers of X from left to right are copied into Z from right to left.

(iii) All odd numbers of Y from left to right are copied into Z from left to right.

(iv) All even numbers of Y from left to right are copied into Z from right to left.

4. A 2-D array defined as A[4..7, -1..3] requires 2 words of storage space for each element. If the array is stored in row major form, calculate the address of A[6,2] given the base address is 100.

5. An array Arr[50][100] is stored in the memory along the row with each element occupying 2 bytes.

Find out the address of the location Arr[20][50], if the location Arr[10][25] is stored at the address 1000.

6. Write a function in C++ which accepts an integer array and its size as arguments and arrange all the odd numbers in the first row and even numbers in the second row of a two dimensional array. The unused cells of two dimensional array must be filled with 0.

If the array is 1, 2, 3, 4, 5, 6

The resultant 2-D array is given below

1	3	5	0	0	0
0	0	0	6	4	2

7. Define a function swapcol() in C++ to swap the first column elements with the last column elements, for a two dimensional integer array passed as the argument to the function.

8. An array S[40][30] is stored in the memory along the row with each of the element occupying 2 bytes, find out the memory location for the element S[20][10], if the Base Address of the array is 5000.

9. Write a user defined function named upper_half() which takes a two dimensional array A, with size N rows and N columns as argument and print the upperhalf of the array.

10. Write a function in C++ which accepts a 2-D array of integers and its size as arguments and prints no of even numbers and odd numbers in each column.

If the array is

11	12	31	41
52	62	71	82
9	10	11	12

The output will be

Column 1:	Even numbers :	1	Odd numbers :	2
Column 2:	Even numbers :	3	Odd numbers :	0
Column 3:	Even numbers :	0	Odd numbers :	3
Column 4:	Even numbers :	2	Odd numbers :	1

11. The numbers (89, 20, 31, 56, 64, 48) are required to be sorted using selection, insertion and bubble sort. Show how the list would appear at the end of each pass.

CHAPTER 10 : STACKS & QUEUES

1. Consider the following program for linked **STACK**:

```
struct NODE
{
    int x;
    float y;
    NODE *next;
};
class STACK
{
    NODE *TOP;;
public :
    STACK() { TOP = NULL; }
    void PUSH( );
    void POP( );
    void Show( );
    ~STACK( );
};
```

Define PUSH() & POP() functions outside the class.

2. Write a C++ function to delete an element from a circular queue containing integers. FRONT and REAR represent the front and rear of the queue.

3. Write functions in C++ to INSERT and DELETE operation in a dynamic Queue containing DVD's information.

```
Struct DVD
{
    long no;
    char title[20];
    DVD *link;
};
```

4. Evaluate the following postfix expression using a stack and show the contents of stack after execution of each operation.

(i) FALSE, NOT, TRUE, AND, TRUE, FALSE, OR, AND

(ii) 32, 4, /, 2, *, 12, 3, -, +

5. Give the postfix form expression for the following.

(i) $X - Y / (Z + U) * V$, (ii) $A - B + C * D ^ E * G / H$

6. Given the following class :

```
char *msg[ ] = {"over flow", "under flow"};
class stack
{
    int top, // the stack pointer
    stk[10]; // the elements
    void error(int e_num)
    {
        cout << msg[e_num]; //report error message
    }
    void init( ) { top = 0} //initialize the stack pointer
    void push(int); // put new value in stk
    void pop( ) ; // get the top value
};
```

Define push(), pop() outside the stack. In your definition take care of overflow and under flow condition.

7. Convert the infix expression : (TRUE && FALSE) || !(TRUE || FALSE)

into POSTFIX expression. Then evaluate the postfix expression showing the status of stack at every step.

Chapter 11 & 12 : DATA BASE CONCEPTS & STRUCTURED QUERY LANGUAGE

1. Define the following:

(i) Domain (ii) Tuple (iii) Relation (iv) Primary key (v) Alternate key (vi) Candidate key

2. What is a view ? How are views different from a table ?

3. What is Cartesian product ? How is it different from Union operation ?

4. Write the SQL commands from (a) to (d) and output from (e) to (h) for the following on the basis of tables GARMENT and FABRIC

Table: GARMENT

GCODE	DESCRIPTION	PRICE	FCODE	READYDATE
10023	PENCIL SKRIT	1150	F03	19-DEC-08
10001	FORMAL SHIRT	550	F01	07-MAR-07
10012	BABY TOP	300	F02	11-SEP-09
10024	EVENING GOWN	545	F04	05-JAN-08
10090	FROCK	575	F01	03-OCT-07

Table: FABRIC

FCODE	TYPE
F04	POLYSTER
F02	COTTON
F03	SILK
F01	TERELENE

(a) To display gcode and description of each garment in descending order of gcode.

(b) To display the details of all the garments which have readydate in between 08-dec-07 and 16-nov-08(inclusive of both dates)

(c) To display the average price of all the garments fcode wise.

(d) To display gcode, description, price from garment table and type from fabric table.

(e) SELECT SUM(PRICE) FROM GARMENT WHERE FCODE='F01';

(f) SELECT DESCRIPTION, TYPE FROM GARMENT, FABRIC WHERE GARMENT.FCODE = FABRIC.FCODE AND GARMENT.PRICE= 1150;

(g) SELECT MAX(PRICE) FROM GARMENT;

(h) SELECT COUNT(FCODE) FROM GARMENT;

5. Consider the following tables PRODUCT and CLIENT. Write SQL commands for the statements (a) to (d) and give outputs for SQL queries (e) to (h).

Table: PRODUCT

P_ID	PRODUCTNAME	MANUFACTURER	PRICE
TP01	TALCOM POWDER	LAK	40
FW05	FACE WASH	ABC	45
BS01	BATH SOAP	ABC	55
SH06	SHAMPOO	XYZ	120
FW12	FACE WASH	XYZ	95

Table: CLIENT

C_ID	CLIENTNAME	CITY	P_ID
01	COSMETIC SHOP	DELHI	FW05
06	TOTAL HEALTH	MUMBAI	BS01
12	LIVE LIFE	DELHI	SH06
15	PRETTY WOMAN	DELHI	FW12
16	DREAMS	BANGALORE	TP01

(a) To display the details of those clients whose city is Delhi.

(b) To display the clientname, city from table client and productname and price from table product with their corresponding matching p_id.

(c) To display the manufacturer of 'FACE WASH'.

(d) To increase the price of all products by 10.

(e) SELECT DISTINCT CITY FROM CLIENT;

(f) SELECT MANUFACTURER, MAX(PRICE), MIN(PRICE), COUNT(*) FROM PRODUCT GROUP BY MANUFACTURER;

- (g) SELECT CLIENTNAME, MANUFACTURER FROM PRODUCT, CLIENT WHERE CLIENT.P_ID = PRODUCT.P_ID;
6. (a) Write the difference between update and alter command..
 (b) Define the term degree and cardinality in a relation.
 (c) What do you mean by data redundancy ?
7. What are the various data models available for database systems ?
8. Consider the following table **CABHUB** and **CUSTOMER** and answer (a) and (b) parts of this question.

TABLE:CABHUB

VCODE	VEHICLENAME	MAKE	COLOUR	CAPACITY	CHARGES
100	Innova	Toyota	WHITE	7	15
102	SX4	Suzuki	BLUE	4	14
104	C Class	Mercedes	RED	4	35
105	A-Star	Suzuki	WHITE	3	14
108	Indigo	Tata	SILVER	3	12

TABLE:CUSTOMER

CCODE	CNAME	VCODE
1	Hemant Sahu	101
2	Raj Lal	108
3	Feroza Shah	105
4	Ketan Dhal	104

- (a) Write SQL commands for the following statements:
- To display the names of all the white coloured vehicles.
 - To display name of vehicle, make and capacity of vehicles in ascending order of their sitting capacity.
 - To display the highest charges at which a vehicle can be hired from CABHUB.
 - To display the customer name and the corresponding name of the vehicle hired by them.
- (b) Give the output of the following SQL queries:
- SELECT COUNT(DISTINCT MAKE) FROM CABHUB;
 - SELECT MAX(CHARGES), MIN(CHARGES) FROM CABHUB;
 - SELECT COUNT(*), MAKE FROM CABHUB;
 - SELECT VEHICLENAME FROM CABHUB WHERE CAPACITY = 4;

CHAPTER 13: BOOLEAN ALGEBRA

- 1 (a) State and prove INVOLUTION LAW of Boolean algebra.
- (b) Implement the following Boolean expression.
 $Y = A.B.C + A.B'.C' + A'.B.C + A'.B'.C'$ with minimum number of gates.
- (c) Convert the following Boolean expression into its equivalent Canonical **Sum of Products form (SOP)**:
 $(X+Y+Z) (X+Y+Z') (X'+Y+Z) (X'+Y'+Z')$
- (d) Find POS of the following :
 $F(A,B,C,D) = \sum(0,1,2,4,5,8,9,10,11)$
- (e) Express $F(x,y,z) = x + y'z$ in SOP form.
- (f) Obtain a simplified form for a Boolean expression.
 $F(u,v,w,z) = \sum(0,1,3,4,5,7,8,9,11,12,13,15)$ using karnaugh map.
- (g) A combinational circuit having 4 input and one output produce 1 when
 - (1) All inputs are 1.
 - (2) none of the inputs are 1.
 - (3) odd number of 1 inputs.

Draw the truth table and represent it in SOP and POS form

- (h) There are four parallel railway track at a place. It is desired to design a logic circuit which can give a signal when three or more train together at any given time.
 - (a) Draw a truth table for the above table.
 - (b) Simplify the expression using k-Map.
- (i) Implement the following Boolean expression using NAND gates only.
 $Y = A.B + A.B' + A'.B + A'.B'$
- (j) Write the dual of the following expression.
 $X + X'Y = X$

2. State and prove the following theorems

- a) DeMorgans' Theorem b) Absorption Theorem c) Indempotent law

3. What is duality principle. Give example

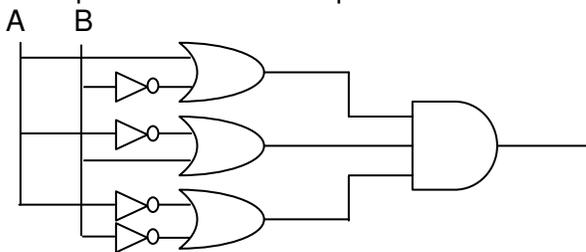
4. Simplify using K-map

$$f(A,B,C,D) = \prod (0,1,3,5,7,10,11,13,15)$$

$$f(P,Q,R,S) = \sum (1,3,4,6,8,11,13,14,15)$$

5. State and verify Associative Law in Boolean algebra.

6. Write the equivalent Boolean expression for the following Logic Circuit.



7. Find the complement of $(AB)' + (A'B)'$

8. Obtain a simplified form for the boolean expression given below (using K-map):

$$F(a, b, c, d) = \pi(0, 1, 2, 4, 5, 8, 10, 12, 13)$$

9. Draw the logic circuit diagram of the following expression using only NAND gates only.

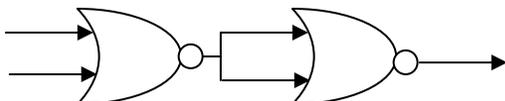
$$a + b'c + c'd'$$

10. Give the duals of $XYZ' + (XY' + 1)$

11. Convert $X + X'Y + X'Y'$ to canonical SOP form.

12. Verify $XY'Z + XY'Z' + X'Y'Z = XY' + Y'Z$ algebraically.

14. Given the following circuit.



What is the output if (i) both inputs are FALSE (ii) one is FALSE and the other is TRUE.

15. (a) Implement the following Boolean expression using NOR gates only.

$$F(A, B, C) = A.B'.C + A'.B.C + A'.B.C'$$

(b) Find the complement of $(A+B)' + (A'+B)'$

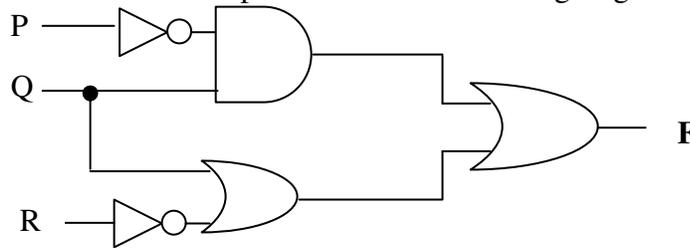
(c) Write SOP and POS forms of function $F(A, B, C)$ whose truth table is given below.

A	B	C	F
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

(d) Obtain a simplified form for the boolean expression given below(using K-map):

$$F(a, b, c, d) = M1. M5. M8 . M12$$

(e) Write the equivalent Boolean Expression for the following Logic Circuit



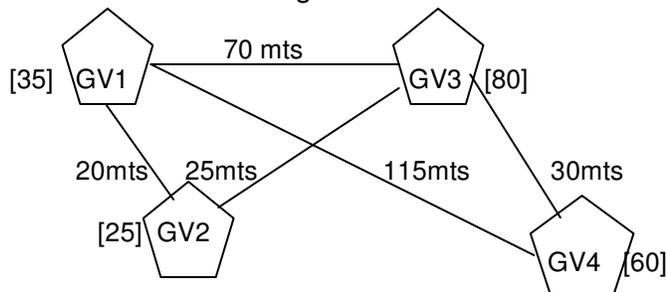
(f) Convert $X(X+Y)$ to canonical POS form.

(g) Reduce the following Boolean expression using K-map

$$F(A,B,C,D) = \Sigma(1,3,4,5,7,9,11,12,13,14)$$

CHAPTER 14 : COMMUNICATION AND OPEN SOURCE CONCEPTS

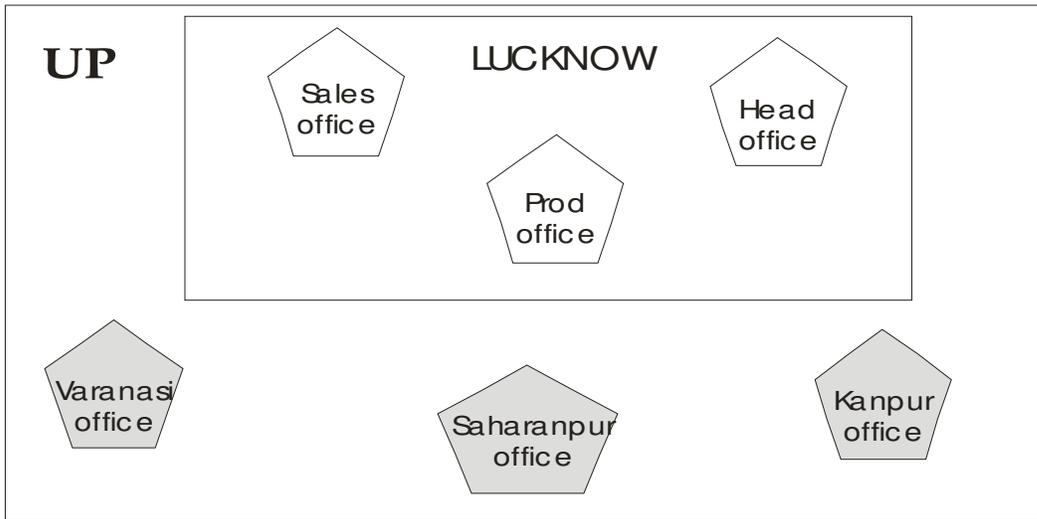
1. How firewall protect our Network?
2. Expand the following:
SMS, URL, WLL, PPP, TCP/IP, FSF, FLOSS, FTP, SMTP, POP, SLIP, FTP, OSS
3. Write one advantage and disadvantage of various network topologies
4. Write short notes for the following
Hub, switch, server, modem, repeater
- 5.(a) What is a proprietary software ?
(b) What is NFS ? What is its purpose ?
(b) What is a backbone network ?
(c) Name one server side scripting language and one client side scripting language?
(d) What are the components required for Networking ?
(e) What are the various physical media used for the data transmission in a Network system.
(f) What is 80 – 20 rule of network design ?
- 6.



Global Village Enterprises has following four buildings in Hyderabad city(As above). Computers in each building are networked but buildings are not networked so far. The company has now decided to connect buildings also.

- (i) Suggest a cable layout for these buildings.
 - (ii) In each of the buildings the management wants that each LAN segment gets a dedicated bandwidth i.e. bandwidth must not be shared. How can this be achieved.
 - (iii) The company also wants for make available shared Internet access for each of the building. How can this be achieved ?
 - (iv) The company wants to link its head office in GV1 building to its another office in Japan. Which type of transmission medium is appropriate for such a link?
What type of network would this connection result into?
7. What are the different switching techniques employed to provide communication between computers ?
 8. What is a repeater ?
 9. What is RJ-45 ?
 10. Write the advantages and disadvantages of Star, Bus topology.
 11. Define the term Firewall.
 12. What is the purpose of using a Web browser ? Name any two commonly used web Browser.
 13. What are the facilities provided by the SERVER in a Network environment ? Explain about different types of server.
 14. What are Spam ? How can you avoid spam ?
 15. What are OSS and FLOSS ?
 16. What is a network ? Why is it needed ?
 17. What are cookies ?
 18. How is a hacker different from a crackers ?
 19. What do you understand by Cyber Law ?
 20. What is Web Hosting ?
 21. How Trojan Horses are different from Worms? Mention any one difference.
 22. Name one server side scripting language and one client side scripting language.
 23. What are Bridge, Gateway and Router ?
 24. Differentiate between Freeware and Shareware?
 25. What is Ethernet ?

26. “Kanganalay Cosmetics” is planning to start their offices in four major cities in Uttar Pradesh to provide cosmetic product support in its retail fields. The company has planned to set up their offices in Lucknow at three different locations and have named them as “Head office”, “Sales office”, & “Prod office”. The company’s regional offices are located at Varanasi, Kanpur & Saharanpur. A rough layout of the same is as follows.



An approximate **distance** between these offices as per network survey team is as follows:

Place from	Place to	Distance
Head office	Sales office	15 KM
Head office	Prod office	8 KM
Head office	Varanasi Office	295 KM
Head office	Kanpur Office	195 KM
Head office	Saharanpur office	408 KM

Number of computers :

Head office	156
Sales office	25
Prod office	56
Varanasi Office	85
Kanpur Office	107
Saharanpur office	105

- (i) Suggest the cable layout(s) for connecting the offices.
- (ii) Name the branch where the **server** should be installed. Justify your answer.
- (iii) Suggest the **device** to be procured by the company for connecting all the computers within each of its offices out of the following devices:
Modem Telephone Switch/Hub
- iv) The company is planning to link its head office situated in Lucknow with the office at Saharanpur. Suggest an economic way to connect it; the company is ready to compromise on the speed of connectivity. Justify your answer.

ALL THE BEST