Which C++ header file(s) will be essentially required to be included to run /execute the following C++ code:

```cpp
void main()
{
    char Msg[] = "Sunset Gardens"; for (int I=5; I<strlen(Msg); I++)
        puts(Msg);
}
```

**Ans:** stdio.h, string.h

Name the header files that shall be need for the following code: (CBSE 2012)

```cpp
void main()
{
    char text[] = "Something"
    cout<"Remaining SMS chars: "<<160-strlen(text)<<endl;
}
```

**Ans:** iostream.h/iomanip.h, string.h

---

2 Marks questions:

1) **Rewrite the following program after removing the syntactical error(s) if any. Underline each correction.**

   **CBSE 2012**
   ```cpp
   #include<iostream.h> Class Item
   {
   long IId, Qty;
   public:
   void Purchase { cin>>IId>>Qty; }
   void Sale()
   {
   cout<<setw(5)<<IId<<"Old:"< Qty<<endl;
   cout<< "New :"<Qty<<endl;
   }
   void main()
   {
   Item I;
   Purchase();
   I.Sale()
   }
   ```

   **Ans:** #include<iostream.h> class Item // C capital
   ```cpp
   {
   long IId, Qty;
   public:
   void Purchase ( ) { cin>>IId>>Qty; } // () after function name
   void Sale( )
   {
   cout<<setw(5)<<IId<<"Old:"< Qty<<endl;
   cout<< "New :"<Qty<<endl;
   }
   ```

   Either the statement is removed or header file included as #include<iomanip.h>
void main()
{
    Item I;
    I. Purchase();  // object missing
    I. Sale();      // ; is missing
}

2) Find the output of the following program:

```
#include<iostream.h>
#include<ctype.h> typedef char Str80[80]; void main()
{char *Notes;
 Str80 str= "vR2GooD"; int L=6;
 Notes =Str;
 while( L>=3)
 { 
    Str[L]=(isupper(Str[L])? tolower(Str[L]) :
           toupper(Str[L])); cout<<Notes<<endl;
    L--;
    Notes ++;
    ;
}}
```

Ans : vR2Good
R2GoOd
2GOOd gOOd

3) Observe the following program and find out, which output(s) out id (i) to (iv) will not
be expected from program? What will be the minimum and maximum value assigned to
the variables Chance?

```
#include<iostream.h> CBSE 2012 #include<stdlib.h>

void main()
{
    randomize();
    int Arr[] = {9,6};, N;
    int Chance = random(2)+10;
    for(int c=0;c<2;c++)
    {
        N= random(2);
        cout<<Arr[N];
    }
}
```

 Ans: The output not expected from program are (i),(ii) and (iv)
Minimum value of Chance =10
Maximum value of Chance = 11
4) **Find the output of the following program:**

```cpp
#include<iostream.h>
class METRO
{
    int Mno, TripNo,
    PassengerCount; public:
    METRO(int Tmno=1) { Mno=Tmno; PassengerCount=0;}
    void Trip(int PC=20) { TripNo++, PassengerCount+=PC;}
    void StatusShow()
    {
        cout<<Mno<<";"<<TripNo<<";"<<PassengerCount<<endl;}
};
void main()
{
    METRO M(5),
    T; M.Trip();
    M.StatusShow();
    T.StatusShow();
    M.StatusShow();
}
Ans : 5: 1: 20
     1: 1: 50
     5: 2: 5
```
2& 3 marks practice questions:
5) Rewrite the following program after removing the syntactical error(s) if any. Underline each correction.

```c
#include<iostream.h>
void main( )
{ F = 10, S = 20;
  test(F;S);
  test(S);
}
void test(int x, int y = 20) { x=x+y;
  count<<x>y;
}
```

6) Rewrite the following program after removing syntactical error(s) if any. Underline each correction.

```c
#include "iostream.h"
Class MEMBER
{ int Mno;
  float Fees;
  PUBLIC:
    void Register ( ) {cin>>Mno>>Fees;}
    void Display( ) {cout<<Mno" : "Fees<<endl;}
  }
void main()
{ MEMBER delete;
  Register();
  delete.Display();
}
```

7) Find the output for the following program:

```c
#include<iostream.h>
#include<ctype.h> void
Encript ( char T[] )
{ for( int i=0 ; T[i] != \0 ; i += 2)
  if( T[i] = = 'A' || T[i] = = 'E' )
    T[i] = '#' ;
  else if (islower (T[i]))
    T[i] = toupper(T[i]);
  else
    T[i] = '@';}
void main()
{ char text [] = "SaVE EA rTh in 2012"; encrypt(text);
  cout<<text<<endl;
}
```

8) Find the output of the following program:

```c
#include<iostream.h>
void main( )
{ int U=10,V=20;
  for(int I=1;I<=2;I++)
  { cout<<"[1]"<<U<<"&"<<V "<<endl;
    cout<<"[2]"<<U+2<<"&"<<V 5 "<<endl; } }
9) Rewrite the following C++ program after removing the syntax error(s) if any. Underline each correction. [CBSE 2010]

```cpp
class FLIGHT
{
    Long FlightCode; Char
    Description[25];
public
    void addInfo()
    {
        cin>>FlightCode; gets(Description);
    }
    void showInfo()
    {
        cout<<FlightCode<".":"<<Description<<endl;
    }
};
void main(){
    FLIGHT F;
    addInfo.F();
    showInfo.F;
}
```

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

```cpp
#include<stdlib.h>
#include<iostream.h>
void main()
{
    randomize();
    char City[][10]=
    {"DEL", "CHN", "KOL", "BOM", "BNG"};
    int Fly;
    for(int I=0; I<3;I++)
    {Fly=random(2) + 1;
    cout<<City[Fly]<< "":";
    }
}
```

**Outputs:**
(i) DEL : CHN : KOL:  (ii) CHN: KOL : CHN:
(iii) KOL : BOM : BNG:  (iv) KOL : CHN : KOL:

11) In the following C++ program what is the expected value of Myscore from options (i) to (iv) given below. Justify your answer.

```cpp
#include<stdlib.h>
#include<iostream.h>
void main()
{
    randomize();
    int Score[] = {25,20,34,56,72,63},Myscore;
    cout<<Myscore<<endl;
}
```

i) 25 (ii) 34 (iii) 20 (iv) Garbage Value.
Function overloading in C++

- A function name having several definitions that are differentiable by the number or types of their arguments is known as function overloading.

Example: A same function `print()` is being used to print different data types:

```cpp
#include <iostream.h>

class printData
{
public:
    void print(int i) {
        cout << "Printing int: " << i << endl;
    }

    void print(double f) {
        cout << "Printing float: " << f << endl;
    }

    void print(char* c) {
        cout << "Printing character: " << c << endl;
    }
};

int main(void)
{
    printData pd;

    // Call print to print integer
    pd.print(5);
    // Call print to print float
    pd.print(500.263);
    // Call print to print character
    pd.print("Hello C++");

    return 0;
}
```

When the above code is compiled and executed, it produces following result:

Printing int: 5
Printing float: 500.263
Printing character: Hello C++
Q 1) Define a class TAXPAYER in C++ with following description:

Private members:
- Name of type string
- PanNo of type string
- Taxabincm (Taxable income) of type float
- TotTax of type double
- A function CompTax( ) to calculate tax according to the following slab:

<table>
<thead>
<tr>
<th>Taxable Income</th>
<th>Tax%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Up to 160000</td>
<td>0</td>
</tr>
<tr>
<td>&gt;160000 and &lt;=300000</td>
<td>5</td>
</tr>
<tr>
<td>&gt;300000 and &lt;=500000</td>
<td>10</td>
</tr>
<tr>
<td>&gt;500000</td>
<td>15</td>
</tr>
</tbody>
</table>

Public members:
- A parameterized constructor to initialize all the members
- A function INTAX( ) to enter data for the tax payer and call function CompTax( ) to assign TotTax.
- A function OUTAX( ) to allow user to view the content of all the data members.

Ans.
class TAXPAYER
{
    char Name[30],PanNo[30];
    float Taxabincm;
    double TotTax;
    void CompTax()
    { if(Taxabincm >500000)
        TotTax= Taxabincm*0.15;
        else if(Taxabincm>300000)
        TotTax= Taxabincm*0.1;
        Else if(Taxabincm>160000)
        TotTax= Taxabincm*0.05;
        else
        TotTax=0.0; }
public:
TAXPAYER(char nm[], char pan[], float tax, double tax) //parameterized constructor
{ strcpy(Name,nm);
    strcpy(PanNo,pan);
    Taxabincm=tax;
    TotTax=ttax; } void
INTAX()
{ gets(Name);
    cin>>PanNo>>Taxabincm;
    CompTax(); } void
OUTAX()
{ cout<<Name<<"\n"<<PanNo<<"\n"<<Taxabincm<<"\n"<<TotTax<<endl; }
};
Q 2 : Define a class HOTEL in C++ with the following description:

Private Members
- Rno //Data Member to store Room No
- Name //Data Member to store customer Name
- Tariff //Data Member to store per day charge
- NOD //Data Member to store Number of days
- CALC //A function to calculate and return amount as NOD*Tariff and if the value of NOD*Tariff is more than 10000 then as 1.05*NOD*Tariff

Public Members:
- Checkin( ) //A function to enter the content RNo,Name, Tariff and NOD
- Checkout() //A function to display Rno, Name, Tariff, NOD and Amount (Amount to be displayed by calling function CALC( ))

Solution :
```
#include<iostream.h>
class HOTEL
{ unsigned int Rno; char
    Name[25];
    unsigned int Tariff;
    unsigned int NOD;
    int CALC()
    { int x; x=NOD*Tariff;
        if( x>10000)
            return(1.05*NOD*Tariff);
        else
            return(NOD*Tariff);
    }
public:
    void Checkin()
    {cin>>Rno>>Name>>Tariff>>NOD;}
    void Checkout()
    {cout<<Rno<<Name<<Tariff<<NOD<<CALC();}
};
```

Q 3 Define a class Applicant in C++ with following description:

Private Members
- A data member ANo (Admission Number) of type long
- A data member Name of type string
- A data member Agg(Aggregate Marks) of type float
- A data member Grade of type char
- A member function GradeMe( ) to find the Grade as per the Aggregate Marks obtained by a student. Equivalent Aggregate marks range and the respective Grades are shown as follows

<table>
<thead>
<tr>
<th>Aggregate Marks</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;= 80</td>
<td>A</td>
</tr>
<tr>
<td>Less than 80 and &gt;= 65</td>
<td>B</td>
</tr>
<tr>
<td>Less than 65 and &gt;= 50</td>
<td>C</td>
</tr>
<tr>
<td>Less than 50</td>
<td>D</td>
</tr>
</tbody>
</table>
Public Members

- A function Enter( ) to allow user to enter values for ANo, Name, Agg & call function GradeMe( ) to find the Grade
- A function Result ( ) to allow user to view the content of all the data members.

Ans:
class Applicant
{
long ANo;
char Name[25];
float Agg;
char Grade;

void GradeMe( )
{
if (Agg >= 80)
    Grade = ‘A’;
else if (Agg >= 65 && Agg < 80)
    Grade = ‘B’;
else if (Agg >= 50 && Agg < 65)
    Grade = ‘C’;
else
    Grade = ‘D’;
}

public:
void Enter ( )
{
    cout <<”n Enter Admission No. “; cin>>ANo;
    cout <<”n Enter Name of the Applicant “; cin.getline(Name,25);
    cout <<”n Enter Aggregate Marks obtained by the Candidate :“; cin>>Agg;
    GradeMe( );
}

void Result( )
{
    cout <<”n Admission No. “<<ANo;
    cout <<”n Name of the Applicant “<<Name;
    cout<<”n Aggregate Marks obtained by the Candidate. “<< Agg;
    cout<<”n Grade Obtained is “<< Grade ;
}

};

Q 4 Define a class ITEM in C++ with following description:
Private members:
2) Icode of type integer (Item Code)
3) Item of type string (Item Name)
4) Price of type Float (Price of each item)
5) Qty of type integer (Quantity in stock)
6) Discount of type float (Discount percentage on the item)
7) A find function finddisc( ) to calculate discount as per the following rule:
   If Qty <=50 discount is 0%
   If 50 < Qty <=100 discount is 5%
   If Qty>100 discount is 10%
Public members:
A function Buy( ) to allow user to enter values for Icode, Item,Price, Qty and call function Finddisc ( ) to calculate the discount.
A function showall ( ) to allow user to view the content of all the data members.
Ans : class ITEM
{int Icode,Qty;
 char item[20];
 float price,discount;
 void finddisc();
 public:
 void buy();
 void showall();
};
 void stock::finddisc()
 {If (qty<=50)
 Discount=0;
 Else if (qty> 50 && qty
 <=100) Discount=0.05*price;
 Else if (qty>100)
 Discount=0.10*price;
 }
 void stock::buy()
 {cout<<"Item Code ";cin>>Icode;
 cout<<"Name :");gets(Item);
 cout<<"Price ";cin>>Price;
 cout<<"Quantity :");cin>>Qty;
 finddisc();
 }
 void TEST::DISPTEST()
 {cout<<"Item
 Code ";cout<<Icode;
 cout<<"Name ");cout<<Name;
 cout<<"Price ");cout<<Price;
 cout<<"Quantity ");cout<<Qty;
 cout<<"Discount ");cout<<Discount;
 }

4 marks Practice Problems :

Q 1 Define a class employee with the following specifications :

Private members of class employee
- empno integer
- ename 20 characters
4) basic, hra, da float
- netpay float
- calculate() A function to calculate basic + hra + da with float return type

Public member function of class employee
6) havedata() function to accept values for empno, sname, basic, hra, da and invoke calculate() to calculate netpay.
7) dispdata() function to display all the data members on the screen.

Q2 Define a class Student with the following specifications :

Private members :
- roll_no integer
- name 20 characters
- class 8 characters
- marks[5] integer
- percentage float
• Calculate() a function that calculates overall percentage of marks and percentage of marks.

**public** members :

8) Readmarks() a function that reads marks and invoke the Calculate function.
9) Displaymarks() a function that prints the marks.

<table>
<thead>
<tr>
<th>adno</th>
<th>4 digit admission number</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>20 characters</td>
</tr>
<tr>
<td>marks</td>
<td>an array of 5 floating point values</td>
</tr>
<tr>
<td>average</td>
<td>average marks obtained</td>
</tr>
</tbody>
</table>

Q3 : Define a class **report** with the following specification :

**Private** members :

•

•

•

• getavg() to compute the average obtained in five subjects

**Public** members :

• readinfo() function to accept values for adno, name, marks, and invoke the function getavg().
• displayinfo() function to display all data members on the screen you should give function definitions.

Q4 Declare a class **myfolder** with the following specification :

**Private members of the class**

• Filenames – an array of strings of size[10][25]( to represent all the names of files inside myfolder)
• Availspace – long ( to represent total number of bytes available in myfolder)
• Usedspace – long ( to represent total number of bytes used in myfolder)

**public members of the class**

• Newfileentry() – A function to accept values of Filenames, Availspace and Usedspace from user
• Retavailspace() – A Function that returns the value of total Kilobytes available ( 1 Kilobytes = 1024 bytes)
• Showfiles() – a function that displays the names of all the files in myfolder

**2 Marks Practice Problems**

1. What is relation between class and object?
2. What are inline functions? Give example
3. Difference between private & public access specifiers.
4. How class implements data-hiding & encapsulation?
5. What is the difference between structure and a class ?
6. How is inline function different from a normal function ?
CONSTRUCTORS AND DESTRUCTORS

1 & 2 Marks Solved Problems:

Q1 :- Answer the questions after going through the following class.
class Exam
{char Subject[20] ;
 int Marks ;
public :
 Exam() // Function 1
{strcpy(Subject, “Computer” ) ; Marks = 0 ;}
 Exam(char P[ ]) // Function 2
{strcpy(Subject, P) ;
 Marks=0 ;
}
 Exam(int M) // Function 3
{strcpy(Subject, “Computer”) ; Marks = M ;}
 Exam(char P[ ], int M) // Function 4
{strcpy(Subject, P) ; Marks = M ;}
};

a) Which feature of the Object Oriented Programming is demonstrated using Function 1, Function 2, Function 3 and Function 4 in the above class Exam? Ans:- Function Overloading (Constructor overloading)

b) Write statements in C++ that would execute Function 3 and Function 4 of class Exam.
Ans:- Exam a(10); and Exam b(“Comp”, 10);

Q2 Consider the following declaration:
class welcome
{public:
 welcome (int x, char ch); // constructor with parameter
 welcome(); // constructor without parameter
 void compute();
private:
 int x; char ch;
};

which of the following are valid statements
welcome obj (33, ‘a9’);
welcome obj1(50, ‘9’);
welcome obj3();
obj1= welcome (45,’T’);
obj3= welcome
Valid and invalid statements are

- `welcome obj (33, 'a9');`  
  valid
- `welcome obj1(50, '9');`  
  valid
- `welcome obj3();`  
  invalid
- `obj1= welcome (45, 'T');`  
  valid
- `obj3= welcome;`  
  invalid

**2 Marks Practice Problems**

Q1 What do you understand by constructor and destructor functions used in classes? How are these functions different from other member functions?  

Q2 What do you understand by default constructor and copy constructor functions used in classes? How are these functions different from normal constructors?  

Q3 Given the following C++ code, answer the questions (i) & (ii).

```cpp
class TestMeOut
{
public:
  ~TestMeOut() // Function 1
  { cout << "Leaving the examination hall " << endl; }
  TestMeOut() // Function 2
  { cout << "Appearing for examination " << endl; }
  void MyWork() // Function 3
  { cout << "Attempting Questions " << endl; }
};
```

(i) In Object Oriented Programming, what is Function 1 referred as and when does it get invoked/called?  
(ii) In Object Oriented Programming, what is Function 2 referred as and when does it get invoked/called?  

**INHERITANCE**

**TYPES OF INHERITANCE**

1. **Single class Inheritance:**

   - Single inheritance is the one where you have a single base class and a single derived class.

     ![Diagram](attachment://inclusion.png)

     - **Class A**: a Base class (super)
     - **Class B**: it is a sub class (derived)
2. **Multilevel Inheritance:**

- In multi level inheritance, a subclass inherits from a class that itself inherits from another class.

![Diagram of multilevel inheritance]

- **Class A**
  - it is a Base class (super) of **B**

- **Class B**
  - it is a sub class (derived) of **A** and base class of class **C**

- **Class C**
  - derived class (sub) of class **B**

3. **Multiple Inheritance:**

- In multiple inheritances, a derived class inherits from multiple base classes. It has properties of both the base classes.

![Diagram of multiple inheritance]

- **Class A**
  - Base class

- **Class B**
  - Base class

- **Class C**
  - Derived class
4. Hierarchical Inheritance:
- In hierarchical inheritance, it's like an inverted tree. So multiple classes inherit from a single base class.

![Hierarchical Inheritance Diagram]

5. Hybrid Inheritance:
- It combines two or more forms of inheritance. In this type of inheritance, we can have a mixture of number of inheritances but this can generate an error of using the same name function from no of classes, which will bother the compiler to how to use the functions.
- Therefore, it will generate errors in the program. This has known as ambiguity or duplicity.
- Ambiguity problem can be solved by using virtual base classes

![Hybrid Inheritance Diagram]

4 marks Solved Problems:

Q1. Consider the following declarations and answer the questions given below:

```cpp
class WORLD {
  int H;
  protected:
  int S;
  public:
  void INPUT(int);
  void OUTPUT();
};
class COUNTRY : private WORLD {
  int T;
  protected:
  int U;
  public:
  void INDATA( int, int)
  void OUTDATA();
};
class STATE : public COUNTRY {
  int M;
  public:
  void DISPLAY (void);}
```
(i) Name the base class and derived class of the class COUNTRY.
(ii) Name the data member(s) that can be accessed from function DISPLAY().
(iii) Name the member function(s), which can be accessed from the objects of class STATE.
(iv) Is the member function OUTPUT() accessible by the objects of the class COUNTRY?

Ans

(i) Base class : WORLD
Derived class : STATE
(ii) M.
(iii) DISPLAY(), INDATA() and OUTDATA() (iv) No

Q2. Consider the following declarations and answer the questions given below:
n: class living_being {
  char name[20];
  protected:
  int jaws;
  public:
  void inputdata(char, int); void outputdata();
}
n: class animal : protected living_being {
  { int tail;
  protected:
  int legs;
  public:
  void readdata(int, int);
  void writedata();
};
n: class cow : private animal {
  char horn_size;
  public:
  void fetchdata(char);
  void displaydata();
};

(i) Name the base class and derived class of the class animal.
(ii) Name the data member(s) that can be accessed from function displaydata. (iii)
Name the data member(s) that can be accessed by an object of cow class. (iv) Is
the member function outputdata accessible to the objects of animal class.

Ans

(i) Base class : living_being
Derived class : cow
(ii) horn_size, legs, jaws
(iii) fetchdata() and displaydata() (iv) No

Q3. Consider the following and answer the questions given below:
n: class MNC
{
  char Cname[25]; // Company
  name protected :
  char Hoffice[25]; // Head office
  public :
  MNC( );
  char Country[25];
  void EnterDate( );
  void DisplayData();
};

(i) Name the base class and derived class of the class MNC.
(ii) Name the data member(s) that can be accessed from function displaydata. (iii)
Name the data member(s) that can be accessed by an object of cow class. (iv) Is
the member function outputdata accessible to the objects of animal class.


class Branch : public MNC
{
  long NOE; // Number of employees
  char Ctry[25]; // Country
  protected:
  void Association();
  public:
  Branch();
  void Add();
  void Show();
};
class Outlet : public Branch
{
  char State[25];
  public:
  Outlet();
  void Enter();
  void Output();
};

(i) Which class’s constructor will be called first at the time of declaration of an object of class Outlet?
(ii) How many bytes an object belonging to class Outlet require?
(iii) Name the member function(s), which are accessed from the object(s) of class Outlet.
(iv) Name the data member(s), which are accessible from the object(s) of class Branch.
Ans (i) class MNC (ii) 129
(iii) void Enter(), void Output(), void Add(), void Show(), void EnterData(), void DisplayData().
(iv) char country[25]

Q4 Consider the following and answer the questions given below:
: class CEO {
  double Turnover;
  protected:
  int Noofcomp;
  public:
  CEO();
  void INPUT();
  void OUTPUT();
};
class Director : public CEO {
  int Noofemp;
  public:
  Director(); void INDATA();
  void OUTDATA();
  protected:
  float Funda;
};
class Manager : public Director {
  float Expense;
  public:
  Manager();
  void DISPLAY(void);
};
(i) Which constructor will be called first at the time of declaration of an Manager?
(ii) How many bytes will an object belonging to class Manager require?
(iii) Name the member function(s), which are directly accessible from the object(s) of class Manager.
(iv) Is the member function OUTPUT() accessible by the objects of the class Director?

Ans (i) CEO()
(ii) 16
(iii) DISPLAY(), INDATA(), OUTDATA(), INPUT(), OUTPUT()
(iv) Yes

4 marks Practice Problems:

Q1 :- Consider the following declarations and answer the questions given below:
class vehicle
{int wheels;
protected:
int passenger;
public:
void inputdata(int, int);
void outputdata();
}
class heavyvehicle: protected vehicle {int dieselpetrol;
protected:
int load;
public:
void readdta(int, int);
void writedata();
class bus:private heavyvehicle
{char marks[20];
public:
void fetchdata(char);
void displaydata();
}

(i) Name the class and derived class of the class heavyvehicle.
(ii) Name the data members that can be accessed from function displaydata()
(iii) Name the data members that can be accessed by an object of bus class
(iv) Is the member function outputdata() accessible to the objects of heavyvehicle class.

Q2:- Consider the following declarations and answer the questions given below:
class book
{
char title[20];
char author[20];
int noof pages;
public:
void read();
void show();
}
class textbook: private textbook
{int noofchapters,
noofassignments; protected:
int standard;
void readtextbook();
void showtextbook();
class physicsbook: public
textbook {char topic[20];
public:
    void
    readphysicsbook();
    void
    showphysicsbook();

(i) Name the members, which can be accessed from the member functions of class physicsbook.
(ii) Name the members, which can be accessed by an object of Class textbook.
(iii) Name the members, which can be accessed by an object of Class physicsbook.
(iv) What will be the size of an object (in bytes) of class physicsbook.

Q3 : Answer the questions (i) to (iv) based on the following: class CUSTOMER
{ int Cust_no;
    char
    Cust_Name[20];
    protected:
    void Register();

public:
    CUSTOMER
    R(); void
    Status( );};
class SALESMAN
{ int Salesman_no;
    char
    Salesman_Name[20];
    protected:
    float
    Salary;
    public:
    SALESMAN
    N();
    void Enter( );
    void Show( );};
class SHOP : private CUSTOMER, public SALESMAN
{ char Voucher_No[10];
    char Sales_Date[8];

public :
    SHOP( );
    void Sales_Entry( );
    void Sales_Detail( );};

(i) Write the names of data members, which are accessible from object belonging to class CUSTOMER.
(ii) Write the names of all the member functions which are accessible from object belonging to class SALESMAN.
(iii) Write the names of all the members which are accessible from member functions of class SHOP.
(iv) How many bytes will be required by an object belonging to class SHOP?
2marks Practice Problems:

1. What is access specifier? What is its role?
2. What are the types of inheritance?
3. What is the significance of inheritance?
4. What is the difference between private and public visibility modes?

1 Mark Questions

1. Observe the program segment carefully and answer the question that follows:
   class stock
   {
       int Ino, Qty; char Item[20];
   public:
       void Enter() { cin>>Ino; gets(Item);
       cin>>Qty; } void issue(int Q) { Qty+=Q;}
       void Purchase(int Q) { Qty-=Q; } int GetIno() { return Ino;}
   };
   void PurchaseItem(int Pino, intPQty)
   {
       fstream File;
File.open(“stock.dat”, ios::binary|ios::in|ios::out);
Stock s;
int success=0;
while(success==0 && File.read((char *)&s,sizeof(s))){
    If(Pino==ss.GetIno())
    {
        s.Purchase(PQty);
        ________________ // statement 1
        ________________ // statement 2
        Success++;
    }
}
if (success ==1)
    cout<< “Purchase Updated”<<endl;
else
    cout<< “Wrong Item No”<<endl;
File.close();
}

Ans.1. i) Statement 1 to position the file pointer to the appropriate place so that the data updation is done for the required item.
    File.seekp( File.tellg()-sizeof(stock);)
    OR
    File.seekp(-sizeof(stock),ios::cur);
ii) Statement 2 to perform write operation so that the updation is done in the binary file.
    File.write((char *)&s, sizeof(s));  OR File.write((char *)&s, sizeof(stock));

3 Marks Question

2. Write a function in c++ to search for details (Phoneno and Calls) of those Phones which have more than 800 calls from binary file “phones.dat”. Assuming that this binary file contains records/ objects of class Phone, which is defined below. class Phone CBSE 2012
{
    Char Phoneno[10]; int Calls;
public:
    void Get() {gets(Phoneno); cin>>Calls;}
    void Billing() { cout<<Phoneno<<
    “#”<<Calls<<endl;} int GetCalls() {return Calls;}
};
Ans 2 : void Search()
{
    Phone P;
    ifstream fin;
    fin.open( “Phone.dat”, ios::binary| ios::in);
    while(fin.read((char *)&P, sizeof(P)))
    {
        if(p.GetCalls() >800)
            p.Billing();
    }
    Fin.close(); //ignore
};
3. Write a function in C++ to add new objects at the bottom “STUDENT.DAT”, assuming the binary file is containing the objects of the following class.

```cpp
class STUD
{int Rno;
 char Name[20];
 public:
 void Enter()
 {cin>>Rno;gets(Name);}
 void Display(){cout<<Rno<<Name<<endl; }
};
```

**Ans.3.**

```cpp
void searchbook(int bookno)
{
{ifstream ifile(“BOOK.DAT”,ios::in|ios::binary);
 if(!ifile)
 {cout<<”could not open BOOK.DAT file”; exit(-1);}
 else
 {BOOK b; int found=0;

 while(ifile.read((char *)&b,
 sizeof(b))) {if(b.RBno()==bookno)
 {b.Display(); found=1; break;}
 }
 if(! found)
 cout<<”record is not found
 “; ifile.close();
 }
}
```

4. Given a binary file PHONE.DAT, containing records of the following class type

```cpp
class Phonlist
{
 char name[20];
 char address[30];
 char areacode[5];
 char Phoneno[15];
 public:
 void Register()
 void Show();
 void CheckCode(char AC[])
 {return(strcmp(areacode,AC));
 };
};
```

Write a function TRANSFER() in C++, that would copy all those records which are having areacode as “DEL” from PHONE.DAT to PHONBACK.DAT.

**Ans.4.**

```cpp
void TRANSFER()
{
 ifstream File1,File2;
 Phonelist P;
 File1.open(“PHONE.DAT”, ios::binary|ios::in);
 File2.open(“PHONEBACK.DAT”, ios::binary|ios::OUT)
 while(File1.read((char *)&P, sizeof(P)))
 { if( p.CheckCode(“DEL”))
 File2.write((char *)&P,sizeof(P));
 }
 File1.close();
 File2.close();
}
POINTERs
Solved Questions

Q. 1 How is *p different from **p ?
Ans : *p means, it is a pointer pointing to a memory location storing a value in it. But **p means, it is a pointer pointing to another pointer which in turn points to a memory location storing a value in it.

Q. 2 How is &p different from *p ?
Ans : &p gives us the address of variable p and *p. dereferences p and gives us the value stored in memory location pointed to by p.

Q. 3 Find the error in following code segment :
Float **p1, p2;
P2 = &p1;
Ans : In code segment, p1 is pointer to pointer, it means it can store the address of another pointer variable, whereas p2 is a simple pointer that can store the address of a normal variable. So here the statement p2 = &p1 has error.

Q. 4 What will be the output of the following code segment ?
    char C1 = 'A';
    char C2 = 'D';
    char *i, *j;
    i = &C1;
    j = &C2;
    *j = j;
    cout << C1; Ans : It will print A.

Q. 5 How does C++ organize memory when a program is run ?
Ans : Once a program is compiled, C++ creates four logically distinct regions of memory :
(i) area to hold the compiled program code
(ii) area to hold global variables
(iii) the stack area to hold the return addresses of function calls, arguments passed to the functions, local variables for functions, and the current state of the CPU.
(iv) The heap area from which the memory is dynamically allocated to the program.
Q. 6 Identify and explain the error(s) in the following code segment:

```cpp
float a[] = { 11.02, 12.13, 19.11, 17.41};
float *j, *k; j = a;
k = a + 4;
j = j * 2;
k = k / 2;
```

```
cout << "\n";
```

Ans: The erroneous statements in the code are:

- `j = j * 2;`
- `k = k / 2;`

Because multiplication and division operations cannot be performed on pointer and `j` and `k` are pointers.

Q. 13 How does the functioning of a function differ when

(i) an object is passed by value?
(ii) an object is passed by reference?

Ans:

(i) When an object is passed by value, the called function creates its own copy of the object by just copying the contents of the passed object. It invokes the object’s copy constructor to create its copy of the object. However, the called function destroys its copy of the object by calling the destructor function of the object upon its termination.

(ii) When an object is passed by reference, the called function does not create its own copy of the passed object. Rather it refers to the original object using its reference or alias name. Therefore, neither constructor nor destructor function of the object is invoked in such a case.

2 MARKS PRACTICE QUESTIONS

- Differentiate between static and dynamic allocation of memory.
- Identify and explain the error in the following program:

```cpp
#include<iostream.h>
int main()
{
int x[] = { 1, 2, 3, 4, 5 };
    for (int i = 0; i < 5; i++)
    {
        cout <<
        *x; x++;
    }
    return 0;
}
```

- Give the output of the following:

```cpp
char *s = “computer”;
for (int x = strlen(s) – 1; x >= 0; x–)
{
    for(int y =0; y <= x; y++) cout << s[y];
    cout << endl;
}
```
4. Identify the syntax error(s), if any, in the following program. Also give void main()
   {const int i = 20;
    const int * const ptr = &i; (*ptr++; int j = 15; ptr
    = &j; }

5. What is ‘this’ pointer? What is its significance?

6. What will be the output of following program? #include<iostream.h>
   void main()
   {
     char name1[] = “ankur”; char name2[] = “ankur”; if (name1
     != name2)
      cout << “n both the strings are not equal”; 
    else
      cout << “n the strings are equal”;
   }

7. Give and explain the output of the following code:
   void junk (int, int *);
   int main()
   {
     int i = 6, j = -4;
     junk (i, &j);
     cout << “i = “ << i << “, j = “ << j << “n”;
     return 0;
   }
   void junk(int a, int *b)
   {
     a = a* a;
     *b = *b * *b;
   }

**UNIT-2 DATA STRUCTURES**

**Data Structure Arrays**

**Example 1.**
For a given array A[10][20] is stored in the memory along the row with each of its elements occupying 4
bytes. Calculate address of A[3][5] if the base address of array A is 5000.
Solution:
For given array A[M][N] where M=Number of rows, N =Number of Columns present in the
array address of A[I][J]= base address+(I * N + J)*sizeof(type)
here M=10, N=20, I=3, J=5, sizeof(type)=4 bytes

address of A[3][5] = 5000 + (3 * 20 + 5) * 4
= 5000 + 65*4=5000+260=5260
Example 2.
An array A[50][20] is stored in the memory along the row with each of its elements occupying 8 bytes. Find out the location of A[5][10], if A[4][5] is stored at 4000.

Solution:
Calculate base address of A i.e. address of A[0][0]
For given array A[M][N] where M=Number of rows, N =Number of Columns present in the array address of A[I][J]= base address+(I * N + J)*sizeof(type)
here M=50, N=20, sizeof(type)=8, I=4, J=5
address of A[4][5] = base address +(4*20 +5)*8
4000 = base address + 85*8
Base address= 4000-85*8= 4000-680=3320 Now to find address of A[5][10]
here M=50, N=20, sizeof(type)=8, I=5, J=10
Address of A[5][10] = base address +(5*20 + 10)*8
=3320 + 110*8 = 3320+880 = 4200

As C, C++ supports n dimensional arrays along the row, the address calculation formula can be generalized for n dimensional array as:
For 3 dimentional array A[m][n][p], find address of a[i][j][k]:
Address of a[i][j][k] = base address + ( (I * n + j ) * p + k ) * sizeof(type)

For 4 dimentional array A[m][n][p][q], find address of a[i][j][k][l]:
Address of a[i][j][k][l] = base address + ( ( (I * n + j ) * p + k ) * p + l ) * sizeof(type)

Column-major order is a similar method of flattening arrays onto linear memory, but the columns are listed in sequence. The programming languages Fortran, MATLAB, use column-major ordering. The array if stored contiguously in linear memory with column-major order would look like the following:
1 4 2 5 3 6

The memory offset could then be computed as: offset = row + column*NUMROWS

Address of element A[row][column] can be computed as:
Address of A[row][column]=base address of A + (column*NUMROWS +rows)* sizeof (type)

Where NUMROWS represents the number of rows in the array in this case, 2.

Treating a row-major array as a column-major array is the same as transposing it. Because performing a transpose requires data movement, and is quite difficult to do in-place for non-square matrices, such transpositions are rarely performed explicitly. For example, software libraries for linear algebra, such as the BLAS, typically provide options to specify that certain matrices are to be interpreted in transposed order to avoid the necessity of data movement.
Example 1.
For a given array A[10][20] is stored in the memory along the column with each of its elements occupying 4 bytes. Calculate address of A[3][5] if the base address of array A is 5000.
Solution:
For given array A[M][N] where M = Number of rows, N = Number of Columns present in the array
Address of A[I][J] = base address + (J * M + I) * sizeof(type)
Here M = 10, N = 20, I = 3, J = 5, sizeof(type) = 4 bytes

Address of A[3][5] = 5000 + (5 * 10 + 3) * 4
= 5000 + 53 * 4 = 5000 + 215 = 5215

Example 2.
An array A[50][20] is stored in the memory along the column with each of its elements occupying 8 bytes. Find out the location of A[5][10], if A[4][5] is stored at 4000.
Solution:
Calculate base address of A i.e. address of A[0][0]
For given array A[M][N] where M = Number of rows, N = Number of Columns present in the array
address of A[I][J] = base address + (J * M + I) * sizeof(type)
Here M = 50, N = 20, sizeof(type) = 8, I = 4, J = 5
address of A[4][5] = base address + (5 * 50 + 4) * 8
= 4000 = base address + 254 * 8
Base address = 4000 - 2032 = 1968
Now to find address of A[5][10]
Here M = 50, N = 20, sizeof(type) = 8, I = 5, J = 10
Address of A[5][10] = base address + (10 * 50 + 10) * 8
= 1968 + 510 * 8 = 1968 + 4080 = 6048

4 Marks Questions
1. Write a function in C++ which accepts an integer array and its size as arguments and replaces elements having even values with its half and elements having odd values with twice its value
2. Write a function in C++ which accepts an integer array and its size as argument and exchanges the value of first half side elements with the second half side elements of the array.
   Example: If an array of eight elements has initial content as 2, 4, 1, 6, 7, 9, 23, 10. The function should rearrange the array as 7, 9, 23, 10, 2, 4, 1, 6.
3. Write a function in C++ to find and display the sum of each row and each column of 2-dimensional array. Use the array and its size as parameters with int as the data type of the array.
4. Write a function in C++, which accepts an integer array and its size as parameters and rearrange the array in reverse. Example if an array of five members initially contains the elements as 6, 7, 8, 13, 9, 19
   Then the function should rearrange the array as 19, 9, 13, 8, 7, 6
5. Write a function in C++, which accept an integer array and its size as arguments and swap the elements of every even location with its following odd location. Example: if an array of nine elements initially contains the elements as 2, 4, 1, 6, 5, 7, 9, 23, 10
   Then the function should rearrange the array as 4, 2, 6, 1, 7, 5, 23, 9, 10
6. Write a function in C++ which accepts an integer array and its size as arguments and replaces elements having odd values with thrice and elements having even values with twice its value. Example: If an array of five elements initially contains the elements 3, 4, 5, 16, 9
   Then the function should rearrange the content of the array as 9, 8, 15, 32, 27
STACKS, QUEUES AND LINKED LIST

Stack

Application of stacks in infix expression to postfix expression conversion
Infix expression  operand1 operator operand2 for example a+b
Postfix expression operand1 operand2 operator for example ab+
Prefix expression operator operand1 operand2 for example +ab

Some example of infix expression and their corresponding postfix expression
Infix expression  postfx expression
a*(b-c)/e  abc- *e/
(a+b)*(c-d)/e  ab+cd-*e/
(a+b*c)/(d-e)+f  abc*+de-/f+

Algorithm to convert infix expression to postfix expression using stack
Let the infix expression INEXP is to be converted in to equivalent postfix expression POSTEXP. The postfix expression POSTEXP will be constructed from left to right using the operands and operators (except “(“ and “)”) from INEXP. The algorithm begins by pushing a left parenthesis onto the empty stack, adding a right parenthesis at the end of INEXP, and initializing POSTEXP with null. The algorithm terminates when stack become empty.
The algorithm contains following steps

1. Initialize POSTEXP with null
2. Add ‘)’ at the end of INEXP
3. Create an empty stack and push ‘(‘ on to the stack
4. Initialize i=0,j=0
5. Do while stack is not empty
6. If INEXP[i] is an operand then
   POSTEXP[j]=INEXP[i]
   I=i+1
   J=j+1
   Goto step 5
(iv) If INEXP[i] is ‘(‘ then
    push (INEXP[i])
    i=i+1
    Goto step 5
(v) If INEXP[i] is an operator then
   While precedence of the operator at the top of the stack > precedence of
   operator POSTEXP[j]=pop()
   J=j+1
   End of while
   Push (INEXP[i])
   I=i+1
   Goto step 5
R. If INEXP[i] is ‘)’ then
   While the operator at the top of the stack is not ‘(‘
   POSTEXP[j]=pop()
   J=j+1
   End while
   Pop()
(ii) End of step 5
(iii) End algorithm
For example convert the infix expression \((A+B)*(C-D)/E\) into postfix expression showing stack status after every step.
<table>
<thead>
<tr>
<th>Symbol scanned from infix</th>
<th>Stack status (bold letter shows the top of the stack)</th>
<th>Postfix expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>(</td>
<td>(</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>(</td>
<td>A</td>
</tr>
<tr>
<td>+</td>
<td>(+(</td>
<td>A</td>
</tr>
<tr>
<td>B</td>
<td>(+</td>
<td>AB</td>
</tr>
<tr>
<td>)</td>
<td>(</td>
<td>AB+</td>
</tr>
<tr>
<td>*</td>
<td>*(</td>
<td>AB+</td>
</tr>
<tr>
<td>C</td>
<td>*(</td>
<td>AB+C</td>
</tr>
<tr>
<td>-</td>
<td>(*(-</td>
<td>AB+C</td>
</tr>
<tr>
<td>D</td>
<td>(*(-</td>
<td>AB+CD</td>
</tr>
<tr>
<td>)</td>
<td>*(</td>
<td>AB+CD-</td>
</tr>
<tr>
<td>/</td>
<td>(/</td>
<td>AB+CD-*</td>
</tr>
<tr>
<td>E</td>
<td>(/</td>
<td>AB+CD-*E</td>
</tr>
<tr>
<td>)</td>
<td></td>
<td>AB+CD-*E/</td>
</tr>
</tbody>
</table>

Answer: Postfix expression of \((A+B)*(C-D)/E\) is \(AB+CD-*E/\)

**Evaluation of Postfix expression using Stack**

Algorithm to evaluate a postfix expression P.

1. Create an empty stack
2. \(i=0\)
3. while P[\(i\)] != NULL
   1. if P[\(i\)] is operator then
      1. Push(P[\(i\)])
      2. \(i=i+1\)
   2. Else if P[\(i\)] is a operand then
      1. Operand2=pop()
      2. Operand1=pop()
      3. Push (Operand1 operator Operator2)
4. End if
5. End of while

5. return pop() // return the calculated value which available in the stack.

End of algorithm

**Example:** Evaluate the following postfix expression showing stack status after every step

\[8, 2, +, 5, 3, -, *, 4 /\]
<table>
<thead>
<tr>
<th>token scanned from postfix expression</th>
<th>Stack status (bold letter shows the top of the stack) after processing the scanned token</th>
<th>Operation performed</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>8</td>
<td>Push 8</td>
</tr>
<tr>
<td>2</td>
<td>8, 2</td>
<td>Push 2</td>
</tr>
<tr>
<td>+</td>
<td>10</td>
<td>Op2=pop() i.e 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Op1=pop() i.e 8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Push(op1+op2) i.e. 8+2</td>
</tr>
<tr>
<td>5</td>
<td>10, 5</td>
<td>Push(5)</td>
</tr>
<tr>
<td>3</td>
<td>10, 5, 3</td>
<td>Push(3)</td>
</tr>
<tr>
<td>-</td>
<td>10, 2</td>
<td>Op2=pop() i.e 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Op1=pop() i.e 5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Push(op1-op2) i.e. 5-3</td>
</tr>
<tr>
<td>*</td>
<td>20</td>
<td>Op2=pop() i.e 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Op1=pop() i.e 10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Push(op1-op2) i.e. 10*2</td>
</tr>
<tr>
<td>4</td>
<td>20, 4</td>
<td>Push 4</td>
</tr>
<tr>
<td>/</td>
<td>5</td>
<td>Op2=pop() i.e 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Op1=pop() i.e 20</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Push(op1/op2) i.e. 20/4</td>
</tr>
<tr>
<td>NULL</td>
<td>Final result 5</td>
<td>Pop 5 and return 5</td>
</tr>
</tbody>
</table>

Evaluate the following Boolean postfix expression showing stack status after every step

**True, False, True, AND, OR, False, NOT, AND**

<table>
<thead>
<tr>
<th>token scanned from postfix expression</th>
<th>Stack status (bold letter shows the top of the stack) after processing the scanned token</th>
<th>Operation performed</th>
</tr>
</thead>
<tbody>
<tr>
<td>True</td>
<td>True</td>
<td>Push True</td>
</tr>
<tr>
<td>False</td>
<td>True, False</td>
<td>Push False</td>
</tr>
<tr>
<td>True</td>
<td>True, False, True</td>
<td>Push True</td>
</tr>
<tr>
<td>AND</td>
<td>True, False</td>
<td>Op2=pop() i.e. True</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Op1=pop() i.e. False</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Push(Op2 AND Op1) i.e. False AND True=False</td>
</tr>
<tr>
<td>OR</td>
<td>True</td>
<td>Op2=pop() i.e. False</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Op1=pop() i.e. True</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Push(Op2 OR Op1) i.e. True OR False=True</td>
</tr>
<tr>
<td>False</td>
<td>True, False</td>
<td>Push False</td>
</tr>
<tr>
<td>NOT</td>
<td>True</td>
<td>Op1=pop() i.e. False</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Push(NOT False) i.e. NOT False=True</td>
</tr>
<tr>
<td>AND</td>
<td>True</td>
<td>Op2=pop() i.e. True</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Op1=pop() i.e. True</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Push(Op2 AND Op1) i.e. True AND True=True</td>
</tr>
<tr>
<td>NULL</td>
<td>Final result True</td>
<td>Pop True and Return True</td>
</tr>
</tbody>
</table>
#include <iostream.h>
struct node {
    int item; node *next;};
class queue {
    node *front, *rear; public:
    queue() {front=NULL; rear=NULL;} //constructor to create empty queue
    void addQ(int item);
    int delQ();
};

void queue::addQ(int item) {
    node *t=new node;
    if(t==NULL)
        cout<<"memory not available, queue is full"<<endl;
    else
        {t->item=item; t->next=NULL;
        if (rear==NULL)  //if the queue is empty
            {rear=t; front=t;}  //rear and front both will point to the first node
        else
            {rear->next=t; rear=t;
        }
    }

int queue::delQ()
{
    if(front==NULL)
        cout<<"queue is empty"<<return 0;
    else
        {node *t=front; int r=t->item;
        front=front->next; //move front to the next node of the queue if(front==NULL)
            rear=front; delete t;
        r;
        return r;
    }
}

void main() { queue q1; q1.addQ(3);
q1.addQ(5) ; q1.addQ(7) ;
cout<<q1.delQ()<<endl ;
cout<<q1.delQ()<<endl ;
cout<<q1.delQ()<<endl; cout<<q1.delQ()<<endl;}

Circular queue using array

2,3 & 4 Marks Practice Questions

1. Convert the following infix expressions to postfix expressions using stack
   1. A + (B * C) ^ D – (E / F - G)
   2. A * B / C * D ^ E * G / H
   3. ((A*B)-(C*D)*E/F)*G

2. Evaluate the following postfix expression E given below; show the contents of the stack during
   the evaluation
   1. E= 5,9,+2,/^4,1,1,3,-,*,+  
   2. E= 80,35,20,^-25,5,+-,*  
   3. E= 30,5,2,^12,6,/-,+-  
   4. E=15, 3, 2, +, /, 7, + 2, *

3. An array A[40][10] is stored in the memory along the column with each element occupying 4 bytes.
   Find out the address of the location A[3][6] if the location A[30][10] is stored at the address 9000. 3
4 Define functions in C++ to perform a PUSH and POP operation in a dynamically allocated stack considering the following:

```cpp
struct Node
{ int X,Y;
 Node *Link;
};
class STACK{
 Node *Top;
 public:
 STACK() { TOP=NULL; }
 void PUSH();
 void POP();
 ~STACK();
};
```

5. Write a function in C++ to perform an Add and Delete operation in a dynamically allocated Queue considering the following:

```cpp
struct node
{ int empno ;char name[20] ;float sal ; Node *Link;
};
```

**UNIT-3 DATABASE AND SQL**

1&2 mark questions

Q2. Define the terms:
   i. Database Abstraction
   ii. Data inconsistency
   iii. Conceptual level of database implementation/abstraction
   iv. Primary Key
   v. Candidate Key
   vi. Relational Algebra
   vii. Domain

**Ans:** Define the terms:

1. **Database Abstraction**
   Ans: Database system provides the users only that much information that is required by them, and hides certain details like, how the data is stored and maintained in database at hardware level. This concept/process is Database abstraction.

2. **Data inconsistency**
   Ans: When two or more entries about the same data do not agree i.e. when one of them stores the updated information and the other does not, it results in data inconsistency in the database.

3. **Conceptual level of database implementation/abstraction**
   Ans: It describes what data are actually stored in the database. It also describes the relationships existing among data. At this level the database is described logically in terms of simple data-structures.

4. **Primary Key**
   Ans: It is a key/attribute or a set of attributes that can uniquely identify tuples within the relation.
v. Candidate Key
Ans: All attributes combinations inside a relation that can serve as primary key are candidate key as they are candidates for being as a primary key or a part of it.

vi. Relational Algebra
Ans: It is the collection of rules and operations on relations (tables). The various operations are selection, projection, Cartesian product, union, set difference and intersection, and joining of relations.

vii. Domain
Ans: It is the pool or collection of data from which the actual values appearing in a given column are drawn.

2 marks Practice questions

10) What is relation? What is the difference between a tuple and an attribute?
11) Define the following terminologies used in Relational Algebra:
   (i) selection  (ii) projection  (iii) union   (iv) Cartesian product
   □ What are DDL and DML?
   □ Differentiate between primary key and candidate key in a relation?
   □ What do you understand by the terms Cardinality and Degree of a relation in relational database?
   □ Differentiate between DDL and DML. Mention the 2 commands for each category.

Database and SQL : 6 marks questions

1. Write SQL Command for (a) to (d) and output of (g)

<table>
<thead>
<tr>
<th>TABLE : GRADUATE</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>S.NO</strong></td>
</tr>
<tr>
<td>-------</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>10</td>
</tr>
</tbody>
</table>

□ List the names of those students who have obtained DIV I sorted by NAME.
□ Display a report, listing NAME, STIPEND, SUBJECT and amount of stipend received in a year assuming that the STIPEND is paid every month.
□ To count the number of students who are either PHYSICS or COMPUTER SC graduates.
□ To insert a new row in the GRADUATE table: 11,”KAJOL”, 300, “computer sc”, 75, 1
□ Give the output of following sql statement based on table GRADUATE:
   - Select MIN(AVERAGE) from GRADUATE where SUBJECT="PHYSICS";
   - Select SUM(STIPEND) from GRADUATE WHERE div=2;
   - Select AVG(STIPEND) from GRADUATE where AVERAGE>=65;
   - Select COUNT(distinct SUBJECT) from GRADUATE;

Sol:
(b) SELECT NAME from GRADUATE where DIV = ‘I’ order by NAME;
(c) SELECT NAME,STIPEND,SUBJECT, STIPEND*12 from GRADUATE;
(d) SELECT SUBJECT,COUNT(*) from GRADUATE group by SUBJECT
    having SUBJECT=‘PHYISICS’ or SUBJECT=‘COMPUTER SC’;
(e) INSERT INTO GRADUATE values(11,’KAJOL’,300,’COMPUTER SC’,75,1);
(f)  i) 63
   ii) 800
   iii) 475
   iv) 4

3. Write SQL command for (i) to (vii) on the basis of the table SPORTS

<table>
<thead>
<tr>
<th>Student NO</th>
<th>Class</th>
<th>Name</th>
<th>Game1</th>
<th>Grade</th>
<th>Game2</th>
<th>Grade2</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>7</td>
<td>Sammer</td>
<td>Cricket</td>
<td>B</td>
<td>Swimming</td>
<td>A</td>
</tr>
<tr>
<td>11</td>
<td>8</td>
<td>Sujit</td>
<td>Tennis</td>
<td>A</td>
<td>Skating</td>
<td>C</td>
</tr>
<tr>
<td>12</td>
<td>7</td>
<td>Kamal</td>
<td>Swimming</td>
<td>B</td>
<td>Football</td>
<td>B</td>
</tr>
<tr>
<td>13</td>
<td>7</td>
<td>Venna</td>
<td>Tennis</td>
<td>C</td>
<td>Tennis</td>
<td>A</td>
</tr>
<tr>
<td>14</td>
<td>9</td>
<td>Archana</td>
<td>Basketball</td>
<td>A</td>
<td>Cricket</td>
<td>A</td>
</tr>
<tr>
<td>15</td>
<td>10</td>
<td>Arpit</td>
<td>Cricket</td>
<td>A</td>
<td>Athletics</td>
<td>C</td>
</tr>
</tbody>
</table>

(a) Display the names of the students who have grade ‘C’ in either Game1 or Game2 or both.
(b) Display the number of students getting grade ‘A’ in Cricket.
(c) Display the names of the students who have same game for both Game1 and Game2.
(d) Display the games taken up by the students, whose name starts with ‘A’.
(e) Add a new column named ‘Marks’.
(f) Assign a value 200 for Marks for all those who are getting grade ‘B’ or grade ‘A’ in both Game1 and Game2.

**Ans:**

a) SELECT Name from SPORTS where grade='C' or Grade2='C';
b) SELECT Count(*) from SPORTS where grade='A';
c) SELECT name from SPORTS where game1 = game2;
d) SELECT game,game2 from SPORTS where name like 'A%';
e) ALTER TABLE SPORTS add (marks int(4));
f) UPDATE SPORTS set marks=200 where grade='A';

Consider the following tables Stationary and Consumer. Write SQL commands for the statement (i) to and output for SQL queries (v) to (viii):

<table>
<thead>
<tr>
<th>S_ID</th>
<th>StationaryName</th>
<th>Company</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>DP01</td>
<td>Dot Pen</td>
<td>ABC</td>
<td>10</td>
</tr>
<tr>
<td>PL02</td>
<td>Pencil</td>
<td>XYZ</td>
<td>6</td>
</tr>
<tr>
<td>ER05</td>
<td>Eraser</td>
<td>XYZ</td>
<td>7</td>
</tr>
<tr>
<td>PL01</td>
<td>Pencil</td>
<td>CAM</td>
<td>5</td>
</tr>
<tr>
<td>GP02</td>
<td>Gel Pen</td>
<td>ABC</td>
<td>15</td>
</tr>
</tbody>
</table>
Table: Consumer

<table>
<thead>
<tr>
<th>C_ID</th>
<th>ConsumerName</th>
<th>Address</th>
<th>S_ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Good Learner</td>
<td>Delhi</td>
<td>PL01</td>
</tr>
<tr>
<td>06</td>
<td>Write Well</td>
<td>Mumbai</td>
<td>GP02</td>
</tr>
<tr>
<td>12</td>
<td>Topper</td>
<td>Delhi</td>
<td>DP01</td>
</tr>
<tr>
<td>15</td>
<td>Write &amp; Draw</td>
<td>Delhi</td>
<td>PL02</td>
</tr>
<tr>
<td>16</td>
<td>Motivation</td>
<td>Bangalore</td>
<td>PL01</td>
</tr>
</tbody>
</table>

(i) To display the details of those consumers whose Address is Delhi.
(ii) To display the details of Stationary whose Price is in the range of 8 to 15. (Both Value included)
(iii) To display the ConsumerName, Address from Table Consumer, and Company and Price from table Stationary, with their corresponding matching S_ID.
(iv) To increase the Price of all stationary by 2.
(v) SELECT DISTINCT Address FROM Consumer;
(vi) SELECT Company, MAX(Price), MIN(Price), COUNT(*) from Stationary GROUP BY Company;
(viii) Select StationaryName, Price*3 From Stationary;

5. Consider the following tables GARMENT and FABRIC. Write SQL commands for the statements (i) to (iv) and give outputs for SQL queries (v) to (viii).

Table: GARMENT

<table>
<thead>
<tr>
<th>GCODE</th>
<th>DESCRIPTION</th>
<th>PRICE</th>
<th>FCODE</th>
<th>READYDATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>10023</td>
<td>PENCIL</td>
<td>1150</td>
<td>F03</td>
<td>19–DEC–08</td>
</tr>
<tr>
<td>10001</td>
<td>FORMAL</td>
<td>1250</td>
<td>F01</td>
<td>12–JAN–08</td>
</tr>
<tr>
<td>10012</td>
<td>INFORMAL</td>
<td>1550</td>
<td>F02</td>
<td>06–JUN–08</td>
</tr>
<tr>
<td>10024</td>
<td>BABY</td>
<td>750</td>
<td>F03</td>
<td>07–APR–07</td>
</tr>
<tr>
<td>10090</td>
<td>TULIP</td>
<td>850</td>
<td>F02</td>
<td>31–MAR–07</td>
</tr>
<tr>
<td>10019</td>
<td>EVENING</td>
<td>850</td>
<td>F03</td>
<td>06–JUN–08</td>
</tr>
<tr>
<td>10009</td>
<td>INFORMAL</td>
<td>1500</td>
<td>F02</td>
<td>20–OCT–08</td>
</tr>
<tr>
<td>10007</td>
<td>FORMAL</td>
<td>1350</td>
<td>F01</td>
<td>09–MAR–08</td>
</tr>
<tr>
<td>10020</td>
<td>FROCK</td>
<td>850</td>
<td>F04</td>
<td>09–SEP–07</td>
</tr>
<tr>
<td>10089</td>
<td>SLACKS</td>
<td>750</td>
<td>F03</td>
<td>20–OCT–08</td>
</tr>
</tbody>
</table>

Table: FABRIC

<table>
<thead>
<tr>
<th>FCODE</th>
<th>TYPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>F04</td>
<td>POLYSTER</td>
</tr>
<tr>
<td>F02</td>
<td>COTTON</td>
</tr>
<tr>
<td>F03</td>
<td>SILK</td>
</tr>
<tr>
<td>F01</td>
<td>TERELENE</td>
</tr>
</tbody>
</table>

(i) To display GCODE and DESCRIPTION of a each dress in descending order of GCODE.
(ii) To display the details of all the GARMENTs, which have READYDATE in between 08– DEC–07 and 16–JUN–08 (inclusive of both the dates).
(iii) To display the average PRICE of all the GARMENTs, which are made up of FABRIC with FCODE as F03.
(iv) To display FABRIC wise highest and lowest price of GARMENTs from DRESS
(Dispaly FCODE of each GARMENT along with highest and lowest price)
(v) SELECT SUM (PRICE) FROM GARMENT WHERE FCODE= ‘F01’;
(vi) SELECT DESCRIPTION, TYPE FROM GARMENT, FABRIC WHERE
GARMENT.FCODE = FABRIC.FCODE AND GARMENT. PRICE>=1260;
(vii) SELECT MAX (FCODE) FROM FABRIC;
(viii) SELECT COUNT (DISTINCT PRICE) FROM FABRIC;

6. Consider the following WORKERS and DESIG. Write SQL commands for the statements (i)
to (iv) and give outputs for SQL queries (v) to (vi)

<table>
<thead>
<tr>
<th>W_ID</th>
<th>FIRSTNAME</th>
<th>LASTNAME</th>
<th>ADDRESS</th>
<th>CITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>102</td>
<td>Sam</td>
<td>Tones</td>
<td>33 Elm St.</td>
<td>Paris</td>
</tr>
<tr>
<td>105</td>
<td>Sarah</td>
<td>Ackerman</td>
<td>440 U.S. 110</td>
<td>New York</td>
</tr>
<tr>
<td>144</td>
<td>Manila</td>
<td>Sengupta</td>
<td>24 Friends Street</td>
<td>New Delhi</td>
</tr>
<tr>
<td>210</td>
<td>George</td>
<td>Smith</td>
<td>83 First Street</td>
<td>Howard</td>
</tr>
<tr>
<td>255</td>
<td>Mary</td>
<td>Jones</td>
<td>842 Vine Ave.</td>
<td>Losantiville</td>
</tr>
<tr>
<td>300</td>
<td>Robert</td>
<td>Samuel</td>
<td>9 Fifth Cross</td>
<td>Washington</td>
</tr>
<tr>
<td>335</td>
<td>Henry</td>
<td>Williams</td>
<td>12 Moore Street</td>
<td>Boston</td>
</tr>
<tr>
<td>403</td>
<td>Ronny</td>
<td>Lee</td>
<td>121 Harrison St.</td>
<td>New York</td>
</tr>
<tr>
<td>451</td>
<td>Pat</td>
<td>Thompson</td>
<td>11 Red Road</td>
<td>Paris</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>W_ID</th>
<th>SALARY</th>
<th>BENEFITS</th>
<th>DESIGNATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>102</td>
<td>75000</td>
<td>15000</td>
<td>Manager</td>
</tr>
<tr>
<td>105</td>
<td>85000</td>
<td>25000</td>
<td>Director</td>
</tr>
<tr>
<td>144</td>
<td>70000</td>
<td>15000</td>
<td>Manager</td>
</tr>
<tr>
<td>210</td>
<td>75000</td>
<td>12500</td>
<td>Manager</td>
</tr>
<tr>
<td>255</td>
<td>50000</td>
<td>12000</td>
<td>Clerk</td>
</tr>
<tr>
<td>300</td>
<td>45000</td>
<td>10000</td>
<td>Clerk</td>
</tr>
<tr>
<td>335</td>
<td>40000</td>
<td>10000</td>
<td>Clerk</td>
</tr>
<tr>
<td>403</td>
<td>32000</td>
<td>7500</td>
<td>Salesman</td>
</tr>
<tr>
<td>451</td>
<td>28000</td>
<td>7500</td>
<td>Salesman</td>
</tr>
</tbody>
</table>

(i) To display the content of workers table in ascending order of first name.
(ii) To display the FIRSTNAME, CITY and TOTAL SALARY of all Clerks from the tables workers
and design, where TOTAL SALARY = SALARY + BENEFITS.
(iii) To display the minimum SALARY among Managers and Clerks from the table DESIG.
(iv) Increase the BENEFITS of all Salesmen by 10% in table DESIG.
(v) SELECT FIRSTNAME, SALARY FROM WORKERS, DESIG WHERE DESIGNATION =
‘Manager’ AND WORKERS.W_ID = DESIG.W_ID;
(vi) SELECT DESIGNATION, SUM(SALARY) FROM DESIG GROUP
BY DESIGNATION HAVING COUNT(*)>=2;
UNIT-4 BOOLEAN LOGIC

Basic postulates of Boolean Algebra:

Boolean algebra consists of fundamental laws that are based on theorem of Boolean algebra. These fundamental laws are known as basic postulates of Boolean algebra. These postulates state basic relations in boolean algebra, that follow:

I  If X != 0 then x=1 and If X!=1 then x=0
II OR relations (logical addition)
   0 + 0 = 0
   0 + 1 = 1
   1 + 0 = 1
   1 + 1 = 1
III AND relations (logical multiplication)
   0.0 = 0
   1.0 = 0
   1.1 = 1
IV Complement Rules \( \overline{0} = 1, \overline{1} = 0 \)

Principal of Duality

This principle states that we can derive a Boolean relation from another Boolean relation by performing simple steps. The steps are:-
  1. Change each AND(.) with an OR(+) sign
  2. Change each OR(+) with an AND(.) sign
  3. Replace each 0 with 1 and each 1 with 0

e.g
  0+0=0 then dual is 1.1=1
  1+0=1 then dual is 0.1=0

Basic theorem of Boolean algebra

Basic postulates of Boolean algebra are used to define basic theorems of Boolean algebra that provides all the tools necessary for manipulating Boolean expression.

1. Properties of 0 and 1
   (a) 0+X=X
   (b) 1+X=1
   (c) 0.X=0
   (d) 1.X=X
2. Indempotence Law
   (a) X+X=X
   (b) X.X=X
3. Involution Law

   \((X) = X\)
4. Complementarity Law
   (a) X + \overline{X}=1
   (b) X. \overline{X}=0
5. Commutative Law
   (a) X+Y=Y+X
   (b) X.Y=Y.X
6. Associative Law
   (a) X+(Y+Z)=(X+Y)+Z
   (b) X(YZ)=(XY)Z
7. Distributive Law
8. Absorption Law
(a) $X+XY= X$
(b) $X(X+Y)=X$

Some other rules of Boolean algebra
$X+XY=X+Y$

**Demorgan’s Theorem**
A mathematician named DeMorgan developed a pair of important rules regarding group complementation in Boolean algebra.

**Demorgan’s First Theorem**

![Diagram of Demorgan’s First Theorem](image)

**Demorgan’s Second Theorem**

![Diagram of Demorgan’s Second Theorem](image)

**Low Order Thinking Questions: (Boolean Algebra)**
a) State and verify absorption law in Boolean algebra.
Ans. Absorption Law states that :
   a) $X+XY=X$
   b) $X(X+Y)=X$

b) Verify $X'.Y+X.Y'=(X'+Y').(X+Y)$ algebraically.
Ans. LHS $= X'.Y + X.Y'$
   $= (X'+X) (X'+Y') (Y+X) (Y+Y')$
   $= 1. (X'+Y') (X+Y).1$
   $= (X'+Y') (X+Y)$
   $= RHS$, hence proved

c) Write the equivalent Boolean Expression F for the following circuit diagram :

![Circuit Diagram](image)
Ans.: $A'B + AB + B'C$

d) If $F(P,Q,R,S) = \prod (3,4,5,6,7,13,15)$, obtain the simplified form using K-Map. Ans.: 

\[
\begin{array}{c|cccc}
\text{P} & \text{Q} & \text{R} & \text{S} \\
\hline
0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 \\
0 & 1 & 1 & 0 \\
1 & 0 & 0 & 0 \\
1 & 0 & 1 & 1 \\
1 & 1 & 0 & 0 \\
1 & 1 & 1 & 1 \\
\end{array}
\]

Reduction of groups following the reduction rule:


$= P + Q'$


$= Q' + S'$

Pair = M3.M7

$= P + R' + S'$

Therefore POS of $F(P,Q,R,S) = (P + Q')(Q' + S')(P + R' + S')$

e) $F(a,b,c,d) = \Sigma (0,2,4,5,7,8,10,12,13,15)$

$F(a,b,c,d) = B_1 + B_2 + B_3$

$B_1 = m_0 + m_4 + m_{12} + m_8 = c'd'$

$B_2 = m_5 + m_7 + m_{13} + m_{15} = bd$

$B_3 = m_0 + m_2 + m_8 + m_{10} = b'd'$

$F(a,b,c,d) = c'd' + bd + b'd'$

f) Write the equivalent Boolean expression for the following logic circuit:

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

- Express in the product of sums form, the Boolean function $F(X,Y,Z)$, the truth table for which is given below:
3 Marks Practice Questions

13. If \( F(a,b,c,d) = \Sigma(0,2,4,5,7,8,10,12,13,15) \), obtain the simplified form using K-Map.

14. If \( F(a,b,c,d) = \Sigma(0,3,4,5,7,8,9,11,12,13,15) \), obtain the simplified form using KMap

15. Obtain a simplified form for a boolean expression

\( F(U,V,W,Z) = \pi (0,1,3,5,6,7,10,14,15) \)

Network topologies and types

Topology:
- Topology refers to the way in which the workstations attached to the network are interconnected.

The BUS Topology:
- The bus topology uses a common single cable to connect all the workstations. Each computer performs its task of sending messages without the help of the central server. However, only one workstation can transmit a message at a particular time in the bus topology.

Advantages:
- Easy to connect and install.
- Involves a low cost of installation time.
- Can be easily extended.

Disadvantages:
- The entire network shuts down if there is a failure in the central cable.
- Only a single message can travel at a particular time.
- Difficult to troubleshoot an error.

The STAR Topology:
- A STAR topology is based on a central node which acts as a hub. A STAR topology is common in homes networks where all the computers connect to the single central computer using it as a hub.

Advantages:
- Easy to troubleshoot
- A single node failure does not affect the entire network.
- Fault detection and removal of faulty parts is easier.
- In case a workstation fails, the network is not affected.

Disadvantages:
- Difficult to expand.
- Longer cable is required.
- The cost of the hub and the longer cables makes it expensive over others.
- In case hub fails, the entire network fails.

The TREE Topology:
- The tree topology combines the characteristics of the linear bus and the star topologies. It consists of groups of star – configured workstations connected to a bus backbone cable.

Advantages:
- Eliminates network congestion.
- The network can be easily extended.
- Faulty nodes can easily be isolated from the rest of the network.

Disadvantages:
- Uses large cable length.
- Requires a large amount of hardware components and hence is expensive.
- Installation and reconfiguration is very difficult.

Types of Networks:
LAN (Local Area Network):
- A Local Area Network (LAN) is a network that is confined to a relatively small area. It is generally limited to a geographic area such as writing lab, school or building. It is generally privately owned networks over a distance not more than 5 Km.

MAN (Metropolitan Area Network):
- MAN is the networks cover a group of nearby corporate offices or a city and might be either private or public.
**WAN (Wide Area Network):** These are the networks spread over large distances, say across countries or even continents through cabling or satellite uplinks are called WAN.

**PAN (Personal Area Network):** A Personal Area Network is a computer network individual person. It generally covers a range of less than 10 meters. Personal Area Networks can be constructed with cables or wirelessly.

**Network Protocol**

3. A protocol means the rules that are applicable for a network.  
4. It defines the standardized format for data packets, techniques for detecting and correcting errors and so on.  
5. A protocol is a formal description of message formats and the rules that two or more machines must follow to exchange those messages.  
6. E.g. using library books.

**Types of protocols are:**
4. HTTP  
5. FTP  
6. TCP/IP  
7. SLIP/PPP

- **Hypertext Transfer Protocol (HTTP)** is a communications protocol for the transfer of information on the intranet and the World Wide Web. HTTP is a request/response standard between a client and a server. A client is the end-user; the server is the web site.
- **FTP (File Transfer Protocol)** is the simplest and most secure way to exchange files over the Internet. The objectives of FTP are:
  - To promote sharing of files (computer programs and/or data).  
  - To encourage indirect or implicit use of remote computers.  
  - To shield a user from variations in file storage systems among different hosts.  
  - To transfer data reliably, and efficiently.
- **TCP/IP (Transmission Control Protocol / Internet Protocol)**
  
TCP - is responsible for verifying the correct delivery of data from client to server. Data can be lost in the intermediate network. TCP adds support to detect errors or lost data and to trigger retransmission until the data is correctly and completely received.  

IP - is responsible for moving packet of data from node to node. IP forwards each packet based on a four byte destination address (the IP number). The Internet authorities assign ranges of numbers to different organizations. The organizations assign groups of their numbers to departments. IP operates on gateway machines that move data from department to organization to region and then around the world.

**Telnet**

It is an older internet utility that lets us log on to remote computer system. It also facilitates for terminal emulation purpose. Terminal emulation means using a pc like a mainframe computer through networking.

**Wireless/Mobile Computing**

Wireless communication is simply data communication without the use of landlines. Mobile computing means that the computing device is not continuously connected to the base or central network.  

1. **GSM (Global System for Mobile communication):** it is a leading digital cellular system. In covered areas, cell phone users can buy one phone that will work anywhere the standard is supported. It uses narrowband TDMA, which allows eight simultaneous calls on the same radio frequency.  
2. **CDMA (Code Division Multiple Access):** it is a digital cellular technology that uses spread-spectrum techniques. CDMA does not assign a specific frequency to each user. Instead, every channel uses the full available spectrum.  
3. **WLL (Wireless in Local Loop):** WLL is a system that connects subscribers telephone network using radio signals as a substitute for other connecting media.  
4. **Email (Electronic Mail):** Email is sending and receiving messages by computer.
5. **Chat**: Online textual talk in real time, is called Chatting.

6. **Video Conferencing**: a two way videophone conversation among multiple participants is called video conferencing.

7. **SMS (Short Message Service)**: SMS is the transmission of short text messages to and from a mobile phone, fax machine and or IP address.

8. **3G and EDGE**: 3G is a specification for the third generation of mobile communication of mobile communication technology. 3G promises increased bandwidth, up to 384 Kbps when a device is stationary.

**EDGE (Enhanced Data rates for Global Evolution)** is a radio based high speed mobile data standard.

**Network Security Concepts:**

- **Viruses**: Viruses are programs which replicate and attach to other programs in order to corrupt the executable codes. Virus enters the computer system through an external source and become destructive.
- **Worms**: Worms are also self-replicating programs that do not create multiple copies of itself on one computer but propagate through the computer network. Worms log on to computer systems using the username and passwords and exploit the system.
- **Trojan horse**: Though it is a useful program, however, a cracker can use it to intrude the computer system in order to exploit the resources. Such a program can also enter into the computer through an email or free programs downloaded through the Internet.
- **Spams**: Unwanted e-mail (usually of a commercial nature sent out in bulk)
- **Cookies**: Cookies are the text messages sent by a web server to the web browser primarily for identifying the user.
- **Firewall**: A firewall is used to control the traffic between computer networks. It intercepts the packets between the computer networks and allows only authorized packets to pass.
- **Cyber Law**: Cyber law refers to all the legal and regulatory aspects of Internet and the World Wide Web.
- **Cyber Crimes**: Cyber crime involves the usage of the computer system and the computer network for criminal activity.

**Hacking**: Hacking is an unauthorized access to computer in order to exploit the resources.

**Web Services**:

- **WWW**: The World Wide Web or W3 or simply the Web is a collection of linked documents or pages, stored on millions of computers and distributed across the Internet.
- **HTML (Hyper Text Markup Language)**: HTML is a computer language that describes the structure and behavior of a web page. This language is used to create web pages.
- **XML (eXtensible Markup Language)**: Extensible Markup Language (XML) is a meta language that helps to describe the markup language.
- **HTTP (Hyper Text Transfer Protocol)**: A protocol to transfer hypertext requests and information between servers and browsers.
- **Domain Names**: A domain name is a unique name that identifies a particular website and represents the name of the server where the web pages reside.
- **URL**: The Uniform Resource Locator is a means to locate resources such as web pages on the Internet. URL is also a method to address the web pages on the Internet. There are two types of URL, namely, absolute URL and relative URL.
- **Website**: A collection of related web pages stored on a web server is known as a website.
- **Web browser**: A software application that enables to browse, search and collect information from the Web is known as Web browser.
- **Web Servers**: The web pages on the Internet are stored on the computers that are connected to the Internet. These computers are known as web servers.
- **Web Hosting**: Web Hosting or website hosting is the service to host, store and the World Wide Web.
- **Web Scripting**: The process of creating and embedding scripts in a web page is known as Web Scripting. Types of Scripts:

   (i) **Client Side Scripts**: Client side scripts supports interaction within a webpage. E.g. VB Script, Java Script, PHP (PHP’s Hypertext Preprocessor).
(ii) **Server Side Scripts**: - Server side scripting supports execution at server – end. E.g. ASP, JSP, PHP

**OPEN SOURCE TERMINOLOGIES**

- **Free Software**: The S/W’s is freely accessible and can be freely used changed improved copied and distributed by all and payments are needed to make for free S/W.
- **Open Source Software**: S/w whose source code is available to the customer and it can be modified and redistributed without any limitation. OSS may come free of cost but nominal charges has to pay nominal charges (Support of S/W and development of S/W).
- **FLOSS (Free Libre and Open Source Software)**: S/w which is free as well as open source S/W. (Free S/W + Open Source S/W).
- **GNU (GNU’s Not Unix)**: GNU project emphasize on the freedom and its objective is to create a system compatible to UNIX but not identical with it.
- **FSF (Free Software Foundation)**: FSF is a non-profit organization created for the purpose of the free s/w movement. Organization funded many s/w developers to write free software.
- **OSI (Open Source Initiative)**: Open source software organization dedicated to cause of promoting open source software it specified the criteria of OSS and its source code is not freely available.
- **W3C (World Wide Web Consortium)**: W3C is responsible for producing the software standards for World Wide Web.
- **Proprietary Software**: Proprietary Software is the s/w that is neither open nor freely available, normally the source code of the Proprietary Software is not available but further distribution and modification is possible by special permission by the supplier.
- **Freeware**: Freeware are the software freely available, which permit redistribution but not modification (and their source code is not available). Freeware is distributed in *Binary Form* (ready to run) without any licensing fees.
- **Shareware**: Software for which license fee is payable after some time limit, its source code is not available and modification to the software are not allowed.
- **Localization**: localization refers to the adaptation of language, content and design to reflect local cultural sensitivities e.g. Software Localization: where messages that a program presents to the user need to be translated into various languages.
- **Internationalization**: Opposite of localization.

**OPEN SOURCE / FREE SOFTWARE**

- **Linux**: Linux is a famous computer operating system. popular Linux server set of program – LAMP (Linux, Apache, MySQL, PHP)
- **Mozilla**: Mozilla is a free internet software that includes a web browser an email client an HTML editor IRC client
- **Apache server**: Apache web server is an open source web server available for many platforms such as BSD, Linux, and Microsoft Windows etc.
- Apache Web server is maintained by open community of developers foundation.

- **MYSQL**: MYSQL is one of the most popular open source database system. Features of MYSQL:
  - Multithreading
  - Multi–User
  - SQL Relational Database Server
  - Works in many different platform

- **PostgreSQL**: Postgres SQL is a free software object relational database server. PostgreSQL can be downloaded from [www.postgressql.org](http://www.postgressql.org).

- **Pango**: Pango project is to provide an open source framework for the layout and rendering of internationalized text into GTK + GNOME environment. Pango using Unicode for all of its encoding, and will eventually support output in all the world’s major languages.

- **OpenOffice**: OpenOffice is an office applications suite. It is intended to compatible and directly complete with Microsoft office.
  OOo Version 1.1 includes:
  - Writer (word processor)
  - Calc (spreadsheet)
  - Draw (graphics program) etc

- **Tomcat**: Tomcat functions as a servlet container. Tomcat implements the servlet and the JavaServer Pages. Tomcat comes with the jasper compiler that complies JSPs into servlets.

- **PHP(Hypertext Preprocessor)**: PHP is a widely used open source programming language for server side application and developing web content.

- **Python**: Python is an interactive programming language originally as scripting language for Amoeba OS capable of making system calls.

Tips to solve Questions based on Networking

1. **Where Server should be placed**: Server should be placed in the building where the number of computers is maximum.

2. **Suggest a suitable cable layout of connection**: A suitable cable layout can be suggested in the following two ways:
   (i) **On the Basis of Server**: First the location of the Server is found out. Server is placed in that building where the number of computers are maximum (According to 80 – 20 rule). After finding the server position, each building distance is compared with the Server building directly or indirectly (taking other building in between). The shortest distance is counted whether it is through directly or indirectly.
   (ii) **On the Basis of Distance from each building**: The distance between the each building is compared to all other buildings either directly or indirectly. The shortest distance is counted whether it is directly or through some other building.

3. **Where the following devices be placed**:
   4. **MODEM**:
   5. **HUB / SWITCH**: In all the wings
   6. **BRIDGE**:
   7. **REPEATER**: It is used if the distances higher than 70 m. It regenerates data and voice signals.
   8. **ROUTER**: When one LAN will be connected to the other LAN.
1 and 2 Marks Questions

Q(1) What do you mean by a computer network?
Ans:- Computer network is an interconnection of autonomous computers connected together using transmission media.

Q(2) What is the need for networking the computers?
4. Time saving

Q(3) What is the full form of ARPANET?
Ans:- Advanced Research Projects Agency Network

Q(4) What are various data transmission modes?
Ans:- There are three modes of data transmission
- Simplex
- Half-duplex
- Full-duplex

Q(5) What is the difference between Simplex and half duplex transmission?
Ans:- In simples transmission mode, the data can be transferred in only one direction where as in half duplex transmission mode, the data can be transmitted in both directions but one at a time.

Q(6) What do you mean by MODEM?
Ans:- MODEM stands for MODulatorDEModulator. It is a device that can convert an analog signal into digital signal and vice versa.

Q(7) Define the terms Bandwidth.
Ans:- Bandwidth is the range of frequencies that is available for the transmission of data. Wider the bandwidth of a communication channel, the more data it can transmit in a given period of time.

Q(8) What are various types of transmission media?
Ans:- There are two broad categories of transmission media
- Guided media
- Unguided Media

Q(9) Explain in brief the advantages and disadvantages of Twisted pair Cable.
Ans:- Advantages
- Inexpensive
- Often available in existing phone system
- Well tested and easy to get
Disadvantages
- Susceptible to noise (sound, energy etc.)
- Not as durable as coaxial cable
- Does not support high speed

Q(10) What do you mean by communication protocol?
Ans:- A protocol is a set of rules to enable computers to connect with one another and to exchange information with minimum possible error.
Q(11) List various functions of Communication protocol.
Ans:- Data sequencing, Data Formatting, Flow control, Error Control, Connection Establishment and termination, Data Security

Q(12) List commonly used protocols.
Ans:- HTTP, TCT/IP, FTP, SLIP, PPP, SMTP, POP, ICMP

Q(13) What are the main functions of TCP
Ans:- The TCP does the following activities
- It breaks the data into packets that the network
- Verifies that all the packets arrived at the destination
- Reassembles the data

Q(14) What do you mean by network topology?
Ans:- Topology is how the nodes/computers are interconnected together.

Q(15) List various types of Networks.
Ans:- LAN, MAN, WAN

Q(16) Give names of various networking topologies in LAN.

Q(17) Write two advantages and two disadvantages of STAR topology.
Ans:- Advantages of STAR topology
- It is easy to modify and add new computers to a star network without disturbing the rest of the network.
- Troubleshooting a star topology network is easy
Disadvantages
- All the nodes are dependent on the central system. Hub. Failure of hub result in shutting down of whole of the system
- Long cable length is required

Q(18) What is NFS?
Ans:- NFS stands for Network File System. NFS is a protocol that allows a set of computers to access each others files.

HIGHER ORDER THINKING QUESTIONS

Q.1 What is protocol? How many types of protocols are there?
Ans. When computers communicate each other, there needs to be a common set of rules and instructions that each computer follows. A specific set of communication rules is called a protocol. Some protocol: PPP, HTTP, SLIP, FTP, TCP/IP

Q.2 What is the difference between Networking and Remote Networking?
Ans. The main difference between Networking and Remote Networking, is the network which we use in offices or other places locally such LAN or INTERNET and remote
networking is one which we use TERMINAL Services to communicate with the remote users such WAN.

Q.3 What is point-to-point protocol?
Ans. A communication protocol used to connect computer to remote networking services include Internet Service Providers. In networking, the Point-to-Point protocol is commonly used to establish a direct connection between two nodes. Its primary use has been to connect computers using a phone line.

Q.4 How gateway is different from router?
Ans. A gateway operates at the upper levels of the OSI model and translates information between two completely different network architectures. Routers allow different networks to communicate with each other. They forward packets from one network to another based on network layer information. A gateway can interpret and translate the different protocols that are used on two distinct networks. Unlike routers that successfully connect networks with protocols that are similar, a gateway perform an application layer conversion of information from one protocol stack to another.

Q.5 What is the role of network administrator?
Ans. Basic tasks for which a network administrator may be responsible:
- Setting up and configuring network hardware and software.
- Installing and configuring network media and connections.
- Connecting user nodes and peripherals of all kinds to the network.
- Adding users to and removing users from the network.
- Managing user account.
- Ensuring the security of the network.
- Provide training to the users to utilize the network’s resources.

Q.6 What is the difference between baseband and broadband transmission?
Ans. Baseband is a bi-directional transmission while broadband is a unidirectional transmission. No Frequency division multiplexing possible in base band but possible in broadband.

<table>
<thead>
<tr>
<th>SNo</th>
<th>Baseband</th>
<th>Broadband</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Entire bandwidth of the cable is consumed by a signal</td>
<td>broadband transmission, signals are sent on multiple frequencies, allowing multiple signals to be sent simultaneously.</td>
</tr>
<tr>
<td>2</td>
<td>Digital signals</td>
<td>Analog signals</td>
</tr>
<tr>
<td>3</td>
<td>bi-directional transmission</td>
<td>unidirectional transmission</td>
</tr>
<tr>
<td>4</td>
<td>No Frequency division multiplexing possible</td>
<td>Frequency division multiplexing possible</td>
</tr>
<tr>
<td>5</td>
<td>Uses for short distance</td>
<td>Uses for long distance</td>
</tr>
</tbody>
</table>

Q.8 What is the differences between POP3 and IMAP Mail Server?
Ans. IMAP is a standard protocol for accessing e-mail from a local server. A simpler e-mail protocol is Post Office Protocol 3 (POP3), which download mail to the computer and does not maintain the mail on the server. IMAP, e-mails are stored on the server, while in POP3, the messages are transferred to the client’s computer when they are read.
Q.9 Name different layer of the ISO OSI Model.

Q.10 What is client server architecture?
Ans. To designated a particular node which is well known and fixed address, to provide a service to the network as a whole. The node providing the service is known as the server and the nodes that use that services are called clients of that server. This type of network is called Client-Server Architecture.

Q.11 What is FDM? Give example.
Ans. FDM-Frequency Division Multiplexing is used in analog transmission. It is often used in short distance. It is code transparent and any terminal of the same speed can use the same sub-channel after the sub-channel is established. The best example if FDM is the way we receive various stations in a radio.

Q.12 Describe the following in brief:
   i) MOSAIC
   ii) USENET
   iii) WAIS
Ans. i) MOSAIC: is the program for cruising the internet. The National centre wrote this program for Super Computer application at the university of Illinois. It has a simple window interface, which creates useful hypertext links that automatically perform some of the menu bar and button functions.

   ii) USENET: is the way to meet people and share information. Usenet newsgroup is a special group set up by people who want to share common interests ranging from current topic to cultural heritages.

   iii) WAIS: is a WIDE AREA INFORMATION SERVER.
e) Ravya Industries has set up its new center at Kaka Nagar for its office and web based activities. The company compound has 4 buildings as shown in the diagram below:

Center to center distances between various buildings is as follows:

<table>
<thead>
<tr>
<th>Building Pair</th>
<th>Distance (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harsh Building to Raj Building</td>
<td>50</td>
</tr>
<tr>
<td>Raz Building to Fazz Building</td>
<td>60</td>
</tr>
<tr>
<td>Fazz Building to Jazz Building</td>
<td>25</td>
</tr>
<tr>
<td>Jazz Building to Harsh Building</td>
<td>170</td>
</tr>
<tr>
<td>Harsh Building to Fazz Building</td>
<td>125</td>
</tr>
<tr>
<td>Raj Building to Jazz Building</td>
<td>90</td>
</tr>
</tbody>
</table>

Number of Computers in each of the buildings is follows:

<table>
<thead>
<tr>
<th>Building</th>
<th>Computers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harsh Building</td>
<td>15</td>
</tr>
<tr>
<td>Raj Building</td>
<td>150</td>
</tr>
<tr>
<td>Fazz Building</td>
<td>15</td>
</tr>
<tr>
<td>Jazz Building</td>
<td>25</td>
</tr>
</tbody>
</table>

e1) Suggest a cable layout of connections between the buildings. 

Answer:

Layout 1:

Layout 2:
e2) Suggest the most suitable place (i.e. building) to house the server of this organisation with a suitable reason.
Answer:
The most suitable place / block to house the server of this organisation would be Raj Building, as this block contains the maximum number of computers, thus decreasing the cabling cost for most of the computers as well as increasing the efficiency of the maximum computers in the network.
(1 mark for correct placement)

e3) Suggest the placement of the following devices with justification:
Internet Connecting Device/Modem
Switch
Answer:
Raj Building
In both the layouts, a hub/switch each would be needed in all the buildings, to interconnect the group of cables from the different computers in each block
(½ Mark for placement of each device correctly)

e4) The organisation is planning to link its sale counter situated in various parts of the same city, which type of network out of LAN, MAN or WAN will be formed? Justify your answer.
Answer:
The type of network that shall be formed to link the sale counters situated in various parts of the same city would be a MAN, because MAN (Metropolitan Area Networks) are the networks that link computer facilities within a city.
(½ mark for correct type and ½ mark for correct justification)