Dummy PDF file

SUBJECT :- COMPUTER SCIENCE (083)
CLASS XII

Unit-Wise Marks and Period Distribution

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

UNIT 1: PROGRAMMING IN C++

REVIEW: *C++* covered In Class -XI,
Defining a symbol name using typedef keyword and defining a macro using #define directive; Need

for User defined data type;

Structures:
Defining a Structure, Declaring structure variables, Accessing structure elements, Passing structure to Functions as value and reference argument/parameter, Function returning structure, Array of structures, passing an array of structure as an argument/ a parameter to a function;

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, Members 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(::); Declaration of objects as instances of a class; accessing members from object(s), Array of type class, 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 Inheritance, Base Class, Derived Class, Defining 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;
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, fstream 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()**, **put()**, **getline()** and **close()** functions; Detecting end-of-file (with or without using **eof()** function);
Opening a binary file using **in**, **out**, and **app** modes;
open(), **read()**, **write()** and **close()** functions; Detecting end-of-file (with or without using **eof()** function); **tellg(), tellp(), seekg(), seekp()** functions
Pointers:
Declaration and Initialization of Pointers; 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 structures: Deference operator: *, ->; self referencial structures;

UNIT 2: DATA STRUCTURES

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 sort), concatenation of two linear arrays, merging of two sorted arrays;
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):
Operations on Stack (PUSH and POP) and its Implementation in C++, Converting expressions from INFIX to POSTFIX notation and evaluation of Postfix expression;
Queue: (Circular Array and Linked Implementation):
Operations on Queue (Insert and Delete) and its Implementation in C++.

UNIT 3: DATABASES AND SQL

Database Concepts:
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, CHARACTER, DATE;

SQL commands:

CREATE TABLE, DROP TABLE, ALTER TABLE, UPDATE...SET..., INSERT, DELETE; SELECT, DISTINCT, FROM, WHERE, IN, BETWEEN, GROUP BY, HAVING, ORDER BY;

SQL functions: SUM, AVG, COUNT, MAX and MIN;

UNIT 4: BOOLEAN ALGEBRA

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, Idem potent Law, Distributive Law, Absorption Law, Involution law, DeMorgan's Law and their applications;

Obtaining Sum of Product (SOP) and Product of Sum (POS) form from 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);

Basic Logic Gates (NOT, AND, OR, NAND, NOR) and their use in circuits.

UNIT 5: COMMUNICATION AND NETWORK CONCEPTS

Evolution of Networking: ARPANET, Internet, Interspace;

Different ways of sending data across the network with reference to switching techniques;

Data Communication terminologies: Concept of Channel, Baud, 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, Hub, Switch, Gateway;

Different Topologies- Bus, Star, Tree; Concepts of LAN, WAN, MAN;

Protocol: TCP/IP, File Transfer Protocol (FTP), PPP, Level-Remote Login (Telnet), Internet, Wireless/Mobile Communication, GSM, CDMA, WLL, 3G, SMS, Voice mail, Application Electronic Mail, Chat, Video Conferencing;

Network Security Concepts: Cyber Law, Virus threats and prevention, Firewall, Cookies, Hacking;

WebPages; Hyper Text Markup Language (HTML), eXtensible Markup Language (XML); Hyper Text Transfer Protocol (HTTP); Domain Names; URL; Protocol Address; Website, Web browser, Web Servers; Web Hosting.

Unit – I

Chapter -1

PROGRAMMING IN C++

Review: C++ covered in C++

Q1. What are the limitations of Procedural Programming ?

Ans. Limitation of Procedural Programming Paradigm

1. Emphasis on algorithm rather than data.
2. Change in a datatype being processed needs to be propagated to all the functions that use the same data type. This is a time consuming process.
3. The procedural programming paradigm does not model real world very well.

Q2. Define Class.

Ans. A Class represents a group of objects that share common properties and relationships.

Exp:-
```
Class ab
{
statements;
}
ab obj;
```

Class name is ab
{ } are used to write statements with in it
; is termination symbol of the statement
obj is the object of a Class to access the data members of the class

Q3. What are the features of OOP ?

Ans. Features of OOP

1. **Data Abstraction**

   Abstraction refers to the act of representing essential features without including the background details or explanations (i.e. Hiding of Data)

2. **Encapsulation**

   The wrapping up of data and functions (that operate on the data) into a single unit (called class) in known as **Encapsulation.** As a base Encapsulation is a way to implement **data abstraction.**

3. **Modularity**

   The act of dividing a complete program into different individual components (functions) is called modularity. **It is a property of a system that has been decomposed into a set of cohesive and loosely coupled modules**

4. **Inheritance**

   Inheritance is the capability of one class of things to inherit the **features or data members or properties from another class.**

   The Class whose properties of data members are **inherited,** is called **Base Class or Super Class** and the class that inherits these properties, is called **Derived Class or Sub Class.**

   Exp:- **If Class A** inherits the data of **Class B** then we can say **A is Sub Class and B is Super Class.**

5. **Polymorphism**

   Polymorphism is the ability for a message or data to be **processed in more than one form**. Polymorphism (overloading) is the property by which the same message can be

sent to objects of several different classes. In which the same operation is performed differently depending upon the data type it is working upon.

Q5. What is the difference between keyword and identifier?

Ans. Keyword is a special word that is reserved in C++ and having special meaning and purpose goto, struct, else, break etc.
Identifier is the user defined name given to a part of a program (variable name)

Q6. Describe different types of operators in C++.

Ans. Operators (to perform some computational operations or to perform some specific actions)

In C++ operators are divided into following categories:-
(Airthmatic, I/O, Increment/ Decrement, Relational & Logical Operators)

(a) I / O operators **(**Input /Output Operators)

Input Operator (>>) is used to read a value from standard input.
cin object is used for taking input from the user

Example :- int a;
cin>>a; (we can input integer value)

Output Operator(<<) is used to direct a value to standard output.
cout object is used for taking output on the display device.

Example:- int a;
cin>>a;
cout<<a;

The multiple use of input or output operators in one statement is called **cascading of I/O** operators

like : cin>>a>>b;
cout<<a<<b;

(b) Arithmetic Operators

+, - , * , / and %

(c) Increment/ Decrement Operators

Increment Operator (++)
Decrement Operator (- -)

We can use both the operators in postfix and prefix mode as per the requirements in the program statement

example:- postfix	**prefix**
int a=10;	int a=10;
a++;	++a;
cout<<a;	cout<<a;
output is a=10	**a=11**

Note:- The postfix form of ++, --, operators follows the rule **use-then change.**
The prefix form follows the rules **change then use**

(d) Relational Operators

< , <= , = = , > , >= and !=

(e) Logical Operators

logical OR (||) , logical AND (&&) , logical NOT(!)

(f) Conditional Operator (? , :)

C++ offers a conditional operator (? :) that stores a value depending upon a condition. This operator is ternary operator.

Syntax: - expression1 ? expression2 : expression3

Exp:-
```
int result;
Result = marks>= 50 ? 'P' : 'F' ;
```

(g) Some other operators (sizeof)

sizeof num // (num is variable name)
sizeof (type) // (c++ data type)

(h) Assignment Operator (=)

Example:

```
int total , item;
total = total + item;
total + =item;
```

Q7. What is the use of main function ?

Ans. Use of Main function in C++

Main ( ) is used to compile and execute the program code of C++ it is used with standard library and header files of the OOPS Program

Example

```
int main ( ) // return values with its parameters
void main ( ) // not return any value not return keyword is used

void main (parameter) // at the end of the only return keyword is
                      // used except parameters
```

Q8. What are the different data types in C++ ?

Ans. C++ Data types

Data types are means to identify the type of data and associated operations for handling it.

C++ data types are of two types

(i) **Fundamental Data Types**

(ii) **Derived Data Types**

(i) Fundamental Data Types

Fundamental Data Types are those that are not composed of other data types. There are **five** fundamental data types in C++: **Char, int, float, double & void** that represents character, integer, float.

(ii) Derived Data Type

(a) **Array:** It is a set of homogeneous values under one name (similar data elements). Array can be one dimensional, two dimensional or multi dimensional

Syntax:- datatype arrname [size] // Single dimensional

The data type of array elements is known as the Base Type of the array. **An ARRAY is a collection of variables on the same type that are referenced by a common name.**

Q9. Explain Two dimensional array with an example.
Ans.

Declaration of Two Dimensional Arrary
Syntax : - datatype arrname [size] [size] // Two dimensional

Example:- int num[2][5] (array will execute 2 x 5 = 10 times)

```
int main( )
{
int sales [5][5];
int i,j, total ;
for (i=0; i<5, i++)
{
        total =0;
        cout<< "\n";                    \\(escape sequence)
for (j=0 ; j< 5; j ++)
{
        cin>>sales[i][j];
        total =total + sales [i][j];
}
cout<< " sales is =  "<< total;
}
return 0;
}
```

Q10. What are functions ? Give syntax to define a function.

Ans. Functions

A function is a named unit of a group of program statements. This unit can be invoked from other parts of the program. **A function return values and numbers and arguments as per instructions stored in a function**

Syntax

```
type function-name (parameter list)
{
        body of the function
}
```

if type of a function is declared then it return values

Q11. Define Pointer, Reference and Constant .
Ans.

Pointer: A pointer is a variable that holds a memory address. This address is usually the location of another variable in memory.

Reference: A reference is an alternative name for an object. A reference variable provides an **alias** for a previously **defined variable.**

Constant: to declare constant value

Syntax :- const datatype var_name= value;

```
const int num =10;
```

Q12. Explain any two user defined data types.

Ans. User defined derived data types

Class: A class represents a group of similar objects. A class describes all the properties of a data type and an object is an entity created according to that description

```
class cls_name
{
        statement
};
```

Structure: A structure is a collection of variables of different data types referenced under one name. **Variables** defined under structure called with the help of structure object. **struct keyword is used to define structure**

```
struct stru_name
        {
                type varname;
                type varname;
        };
stru_name obj_name;

cin>>obj_name.varname;
cout<<obj_name.varname;
```

Q13. What do you mean by variable ?

i) **Ans. Variables:** Variables represent named storage locations whose values can be manipulated

Q14. What are the different types of errors in C++.

Ans. Types of errors in C++

Errors may be made during program creation even by experienced programmers. Such types of error are detected by the compiler. Debugging means removing the errors.

The errors are categorized in four types:-

(i) Syntax errors
(ii) Linking errors
(iii) Execution –time errors (Run Time errors)
(iv) Logical errors

Q15. Write the C++ equivalent expressions for the following.
Volume = 3 .1459r^2 h/3

Ans. Volume = 3.1459*r*r*h/3;

Q16. Find the syntax error (s) if if any, in the following program:

```
#include<iostream.h>
int main()
int x;
cin<< x;
for(int y = 0; y < 10; y++)
cout >> x+y;
```

Ans The syntax error are:

1. illegal`<<` operator in cin statement.
2. illegal`>>` operator in cout statement., return value function.

Q17. differentiate between a run-time error and syntax error. Give one example of each.

Ans. While execution, a compiled program may not behave properly because of some errors called run time errors. For example, divide by zero is a run time error. The following program segment will generate such an error.The following program segment will generate such an error

```
while flag
    {
        .
        b = b-1 ;
        term = a/b;
        .
     }
```

Array indicates out of bound, and range errors are other examples of run time errors. A syntax error, on the other hand, is because of misuse of a programming language. Any violation of a grammatical rule of a programming language will produce a syntax error. Such errors are caught by language compiler. The following statement is not systactically correct as far as c++ is concerned .

X= y + z** E;

Q18. What is the difference between an object and a class?

Ans. An object is an identifiable entity with some characteristics and behaviour. It represents an entity that can store data and its associated functions.
A class is a group of objects that share common properties and relationships

Q19. Write a program to input any number and print in reverse order

Ans in this program the number is input by user

```
#include<iostream.h>
#include<conio.h>
void main()
{
 clrscr();
 int n,s=0;
```

```
 cout<<" enter the number:-  ";
 cin>>n;
 while(n>0)
 {
  s=s*10+n%10;
  n=n/10;
 }
 cout<<"revese="<<s;
 getch();
}
```

Q20. Write a program for decalring and calling of function inside <u>main</u> and defining outside the main ()

Ans

```
#include<iostream.h>
#include<conio.h>
void main()
{
 void fact();
 fact();
 getch();          // for freeze the montior
}
void fact()
{
 int i=1,n,fact=1;
 cout<<"enter the number=";
 cin>>n;
 while(i<=n)
 {
   fact=fact*i
   i++;
  }
  cout<<"fact="<<fact;
  getch();
}
```

Q21. Swapping of two numbers using function call by value & call by reference.

Ans

Call by value

```
#include<iostream.h>
#include<conio.h>
main()
{
 clrscr();
  int a,b;
  cout<<"enter the number=";
  cin>>a>>b;
  void swap(int,int);
  swap(a,b);
  cout<<"A="<<a<<endl;
  cout<<"B="<<b;
```

```
  getch();
}
void swap(int a,int b)
{
 int t=a;
 a=b;
 b=t;
 cout<<a<<b;
}
```

Call by reference

```
#include<iostream.h>
#include<conio.h>
void main()
{
void swap( int &a, int &b);              //prototype of a function
int num1,num2;
clrscr();
cout << " enter both numbers:   num1 & num2:";
cin>>num1;
cout<< "\n";
cin>>num2;
cout<< "\n  Before swapping numbers are \n";
cout<< " num1=  "<<num1;
cout<< "\n";
cout<< " num2=  "<<num2;
cout<< "\n";
swap(num1,num2);              //calling of function
cout<< "\n  After swapping numbers are \n";
cout<< " num1=  "<<num1;
cout<< "\n";
cout<< " num2=  "<<num2;
cout<< "\n";
getch( );                     //for freeze the monitor
}
      void swap(int &a, int &b)          //function definition
{
int temp=a;
a=b;
b=temp;
}
```

Q22. Name the header files of following built in functions :

Strcpy(), strcat(),log(), clrscr(),setw(),fabs(),isalnum(),isupper()

Ans.

Strcpy()	**string.h**
Strcat()	**string.h**
log()	**math.h**
clrscr()	**conio.h**
setw	**iomanip.h**
fabs()	**math.h**
isalpnum()	**ctype.h**
isupper()	**ctype.h**

Chapter -2
OBJECT ORIENTED PROGRAMMING

Q1. Give an example of polymorphism.
Ans. Polymorphism (overloading)

Function overloading

If a function name is same and use more than one time with different parameters is called function overloading.

```
void num( )
{
        int a;
        a= 5*10;        }
void num1 (int a)
{
        int b;
        b= a*a;
return;
}
```

At calling time we can also overload the functions

Q2. Write 4 advantages of Object oriented programming.

Ans. Advantages of OOP

1. Re-use of code
2. Based on comprehensive approach
3. Program can be maintained using encapsulation and hiding of data
4. Easy to maintain , cost effective
5. Easy to redesign and easy to extend

Q3. Illustrate the concept of function overloading with the help a function namely Area that return a float type value

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

For example, following code overloads a function area to compute areas of circle, rectangle and triangle.

```
float area (float radius)                    // for computing area of a circle
{
        return 3.14 * radius * radius;
}
flaot area ( float length , float breadth)          // for calculation area
{
        return length * breadth;
}
```

Chapter -3
CLASSES AND OBJECTS

Q1. What is class? What is its need?

Classes

A class is a way to bind the data describing an entity and its associated functions together. In C++ , class makes a data type that is used to create objects of this type.

Need for classes

Classes are needed to represent real-world entities that not only have data type properties (Characteristics) but also have associated operations (their behaviour)

Q2 . Differentiate between local and global class with help of example.

Ans. 1. Global Class

A class is said to be global it its declaration occurs outside the bodies of all functions in a program.

2. Local class

A class is said to be local if its occurs inside the function body

```
# include<iostream.h>
class x                          //x is a global class
{
members variable functions of class;
};

x ob1;                           //global object

int main ( )
{
class num                        // num is local class
        {
        public:
        int a=10;
        void num( )              // constructor (same name of class)
        {
        cout<<num;
        };

        x ob2;                           //local object of class x
        num n1;                          // n1is a class object
        n1.num( );
        }                                // closing of main ()
```

Q3. What do you mean by nested class ?Explain.

Ans. Nested class

Declaration of one class in to another class is called is called nested of a class.

```
#include<iostream.h>
```

```
#include<conio.h>      // for clrscr()

class score
{
int a;              //private member variable

class batesman          // nested class
{
int b;

public:
int c;
void total( )
{
cin>>b>>c;
int sum = b+c;
}
batesman( )
{
c=10;
}
};                  //  end of class batesman
batesman obj2;          // object of class batesman

void second( void)
{
cout<<score :: second()<<endl;
cout<<"A =" <<a
}
        score( )
        {
        a= 25;
        }
};      //end of class score;
void main()
{
        score ab;
        batesman bc;
        score :: batesman cd;
        ab.second();
        bc.total();
        cd.total();
}
```

Q4. What are the advantages and disadvantages of inline functions?

Ans. The main advantages of inline functions is that they save on the overheads of a function call as the function is not invoked, rather its code is replaced in the program.

The major disadvantage of inline functions is that with more function calls, more memory is wasted as for every function call, the same function code is inserted in the program. Repeated occurrences of same function code waste memory space.

Q5. What do you understand by member function?

Ans Member function have full access privilege to both the public and private members of the class. These are defined within the class scope that is they are not visible outside the scope of class.

But not member functions are visible outside the class (ordinary functions)

Q6. What are static class members?

Ans A class can have static data members as well as static member functions The static data members are the class variables that are common for all the objects of the class. Only one copy of static data members is maintained which is shared by all the objects of the class. They are visible only with in the class

Q7. Identify the errors in the following code fragment:

```
class ab
{
        int x;
        static int ctr;
Public:
        Void init(void)         {
        x=ctr=0;                }
static void print(void)         {
        cout<<ctr<<x;           }       };
```

Ans In the above code fragment, a static member function is trying to access a non static member that is 'x' which is the error

The correct code for static member function is

```
        static void print(void)
        {
        cout<<ctr;              }
```

Q8. What is the significance of access specifiers in a class?

Ans A class provides three access specifiers namely private, public and protected

(a) A data member declared as private or protected remains hidden from outside world and it can only be accessed by the member functions of the class.

(b) A data member declared as public is made available to the outside world. That is, it can be accessed by any function, any expression in the program but only by using an object of the same class type.

These access labels enforce data hiding and abstraction also.

Q9 . Write the output of the following code

```
#include<iostream.h>
class counter                   {
private:
        unsigned int count;
public:
        counter()               // constructor
        {
        count=0;        }

        void inc_count( )       // increment in count variable
        {
```

```
                count ++;           }

                int get_count( )    {
                return count;       }
                                    };
void main()
{
counter c1,c2;
cout<< " \n value for c1 is:" <<c1.get_count( );
 cout<< " \n value for c2 is:" <<c2.get_count( );
c1.inc_count();
c2.inc_count();
c2.inc_count();
cout<< "\t c1=" <<c1.get_count();
cout<< "\t c2=" <<c2.get_count();          }
```

Ans The output will be

c1=0 c2=0 c1=1 c2=2

Chapter – 4
CONSTRUCTOR AND DESTRUCTORS

Q1. What is Constructor ?
Ans. Constructors:

A constructor is a special member function of a class that is automatically called, when an object is created of that class. **A member function with the same name as its class is called constructor.**

Q2. What is destructor ? What is its need ?
Ans. Destructor:

A destructor is also a member function whose name is the same as the class name but is **preceded by tilde ('~')**

A destructor takes no arguments and no return types can be specified for it. It is called automatically by the compiler when an object is destroyed. A destructor cleans up the storage (memory area of the object) that is no longer accessible.

Need for Destructors

Allocated resources (like constructor) must be de-allocated before the object is destroyed. It works as a de-allocating and releasing memory area and **it perform our works as a clean up tasks.** Therefore, a destructor is equally useful as a constructor is.

```
class stud
 {
        stud()                      //constructor
        {
        cout << " welcome"
```

```
    }
        ~stud()              //destructor
        {
        }
};
    stud s1;
```

Q1. What are the characteristics of a Constructors?

Ans. The constructor functions have certain special characteristics:-

1. Constructor fuctions are invoked automatically when the objects are created.
2. If a class has a constructor, each object of that class will be initialized before any use is made of the object.
3. No return type can be specified for a constructor
4. They can not be inherited through a derived class can call the base class constructor
5. A constructor may not be static.
6. It is not possible to take the address of a constructor
7. Member functions may be called from within a constructor.

Q2. What do you understand by default constructor?

Ans A default constructor is the one that takes no arguments. It is automatically invoked when an object is created without providing any initial values. In case, the programmer has not defined a default constructor, the compiler automatically generates it.

Q3. 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;
```

Ans. 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
```

Q4 . What is copy constructor?

Ans Copy constructor:

A copy constructor is a constructor that can be used to initialize one object with the values from another object of same class during declaration.

Q5 . What is the relationship between constructor and its class? Define constructor also

Ans Both carry the same name. It is a function that automatically initializes the data members of an object at the time of its creation.

Constructor : It removes the memory held by an object, It is the last member function to be called. Morevoer, it carries the same name of the class with tidle (~) as prefix.

Q6. How many time is the copy constructor called in the following code?

```
Sample func(sample u)
{
        sample v(u);
        sample w=v;
        return w;
}
void main()
{
sample x;
sample y = func(x);
sample z = func(y);
}
```

Ans . 8 times

Q9. What will be output of following?

```
# include<iostream.h>
class play
{
        public:
        play()
        {
        calculate();

        void calculate()
        {
        show();
        cout<<"calculating";
        }
                void show()
                {
                cout<<"welcome in C++";
                }
};
void main()
{
play one;
}
```

Ans. Welcome in C++ calculating

Chapter – 5
INHERITANCE: EXTENDING CLASSES

Q1. What is Inheritance ?

Ans. It is a special feature of OOPS. Inheritance is capability to inherit the properties of one class in to another class.

The derive new class is called **derived class (sub class)** and old class is called based class (**super class).**

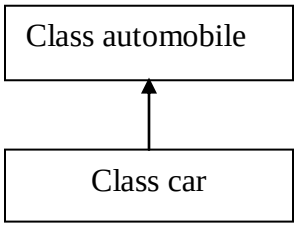
The Class whose properties of data members are **inherited,** is called **Base Class or Super Class** and the class that inherits these properties, is called **Derived Class or Sub Class.**

Exp1:- **If Class A** inherits the data of **Class B** then we can say **A is Sub Class and B is Super Class.**

Q2. What are the different types of inheritance ?

Ans. Type of Inheritance

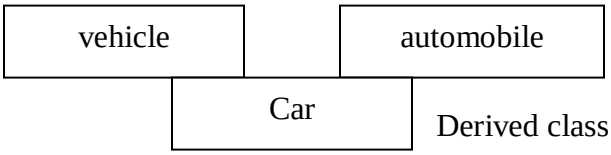
1. Single Inheritance



it is a Base class (super)

it is a sub class (derived)

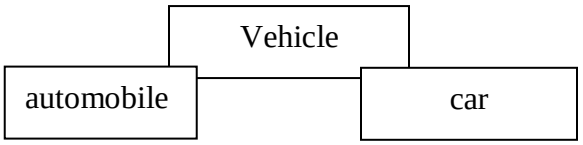
2. Multiple Inheritance



Base class

Derived class

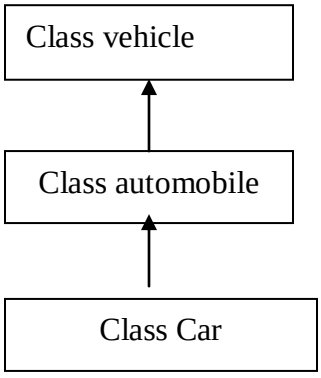
3. Hierarchical Inheritance



Base class (super)

Sub Class(derived)

4. Multilevel Inheritance



it is a Base class (super) of automobile

it is a sub class (derived) of automobile and base class of class car

derived class(sub) of class automobile

5. Hybrid Inheritance

It is combination of two or more forms of inheritance.

Q2. What do you mean by Base and Derived Class ?

Ans. A derived class (or sub class) has to identify the class from which it is derived i.e. its base class (or super class)

```
class derived-class-name : visibility-mode base-class-name
{
    members of derived class;
};
```

class is a key word and **visibility mode is a access speicifier (i.e. public, private or protected), : (colon) is used separation**

colon (:) indicates derived class (sub class) is based on base class(super class)

Example

```
class car : public automobile
{
members
};
```

Q3. What is Multiple Inheritance ?

Ans. Multiple Inheritance means deriving a class from more than one base class.

```
Class kvschool
    {
        int rollno1;
        void num1();
    Public:
        {
        float kvroll;
        void num2();
    };
class kvstud
    {
        int rollno2
        void num3();
    Protected:
        float kvroll1;
        void num4();
    };
class kvclass : public kvschool, public kvstud
    {
        int kvroll2;
    protected:
        void display();
    };

    // to complete the program deifine display () – in class kvclass
        i.e. kvclass ::display()
```

Q4. Define the needs and objectives of Inheritance.

Ans The major needs and objectives of inheritance are:

(i) *It ensures the closeness with the real world models.*
(ii) *It extend the functionality of an existing class.*
(iii) *It establishes "a kind of" relationship.*
(iv) *It helps in reuse of an existing class by a subclass (reusability).*
(v) *It implements transitive nature(if a class Y inherits properties from class X, then all subclassY will automatically inherit the properties of X)*
(vi) *The redundancy can be reduced by abstracting a super class from few sub classes.*
(vii) *It is conept of reusability.*

Q5. Give the following definitions, answer the questions that follow:-

```
#include <iostream.h>
class book
{
char title[20];
char author[20];
int noof pages;
public:
        void read();
        void show();
};
class textbook: private textbook
{
int noofchapters, noof assignments;
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.

Ans

(i) standard , readtextbook(),showtextbook() and topic;
(ii) readtextbook() and showtextbook()
(iii) readphysicsbook(), showphysicsbook(), readtextbook() and showtextbook()
(iv) The size of object of physicsbook= size of book + size of Textbook + size of physicsbook.
= 42+6+20 = 68 bytes

Q6. 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 readdata( 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.**

Ans

(i) **base class = vehicle, derived class = bus**

(ii) The data members passenger, load, make are available to function display data

(iii) No data members can be accessed by the object of bus calss.

(iv) No member functions outputdata () is not accessible to the objects of heavy vehicle class.

Q7 . What type of C++ class members (data members and member functions) are not inherited?

Ans **Data member** : Static data members of the base class are not inherited by the derived class

Member functions: Constructors and destructors of base class are not inherited

Chapter-6

DATA FILE HANDLING

Q1. Define File .

Ans.A file is a bunch of bytes stored on some storage devices.

Q2. How are binary file differ from text files in c++?

Ans When a file is opened in text mode, various character translations may take place, such as the conversion of carriage return and linefeed sequences into newlines. However, no such character translations occur in files opened in binary mode. Any file can be opened in either text or binary mode. The only difference is the occurrence of character translations in text mode.

Q3. What is a stream? Name the streams generally used for file I/O.

Ans A stream is a sequence of bytes. Or in other words, a stream is a flow of bytes into or out of a program. Generally three streams are used for file I/O. These are:

(i) Ifstream: It is derived from istream class and it is used to associate a file with an input buffer so that the file can be read from.

(ii) Ofstream: It is derived from ostream class and it is used to associate a file with an output buffer so that the file can be written from.

(iii) fstream: It is derived from iostream class and it is used to associate a file with a buffer so that the file can be read from as well as written onto.

Q4. Differentiate between ifstream class and ofstream class.

Ans The ifstream class is an input stream class, which provides input operations for file.
The ofstream class is an output stream class, which provides output operations for file.

Q5. Differentiate between functions read() and write().

Ans The read() lets one read a unit of information from a file and a write() lets one write a unit of information to a file.

Q6. Differentiate between getline() and getc() functions.

Ans getc() function can read one character at a time .On the other hand , getline() can read a line of text of specified size.

getc() is defined in stdio.h however getline() is defined in iostream.h

Q7. Name two member functions of ofstream class.

Ans seekp() , tellp()

Q8. Distinguish between ios::out and ios::app.

Ans The ios::out is the default file mode of ofstream. With the mode of the file does not exist, it gets created but if the file exists then its existing contents get deleted. On the other hand ios::app is also an output mode , which creates the file if it does not exist but if the file exists then its existing contents are retained and new information is appended to it.

Q9. What is the difference between the functioning of ios::ate and ios::app file modes.

Ans Both ios::ate and ios::app place the file pointer at the end of the file just opened. The difference between two is that ios::app lets you add data to the end of the file only, while the ios::ate mode when opened with ofstream allows you to write data anywhere in the file, even over old data.

Q10. What is the need and usage of read() and write() functions when there are get() and put() functions for I/O.

Ans The get () and put () function perform I/O byte by byte .On the other hand, read () and write () functions lets you read and write structures and objects in one go without creating the need for I/O for individual constituent fields.

Q11. What is the difference between get() and read()?

Ans The read() declared under iostream.h extracts a given number of characters into an array. The get() declared under iostream.h either extracts the next character or EOF or extracts characters into a char* until eof or delimiter encountered or specified(len-1) bytes have been read.

Q12. Write a C++ program, which reads one line at a time from the disk file TEST.TXT, and displays it to a monitor. Your program has to read all the contents of the file. Assume the length of the line does not exceed 70 characters.

Ans

```
#include<fstream.h>

void main()
{
        char str[80];
        ifstream fin("Test.txt");
        if (!fin)
        {       cout<<"Error in the opening";
                return –1;
        }
        while(fin)
        {       fin.getline(str,70);
                cout<<str<<endl;
        }
        fin.close();
}
```

Q13. Write a function to count the number of blanks present in text file named "PARA.TXT";

Ans

```
        void countblanks()
        {
        char ch;
        int count=0;
        ifstream fin("PARA.TXT",ios::in);
        while(!fin.eof())
        {       fin.get(ch)
                if(fin.eof())
                break;
                if(ch==' ')
                count++;                        }
        fin.close();
        cout<<count;                            }
```

Q14. Write a Program to write and read a structure using write() and read() function using a binary file

```
#include<fstream.h>

#include<string.h>
#include<conio.h>
struct customer
{
char name[51];
float balance;          };
```

```
void main()
{
clrscr();
customer savac;
strcpy(savac.name,"Tina Marshall");
savac.balance=21310.75;
ofstream fout;
fout.open("Saving" , ios::out|ios::binary);
if(!fout)
{          cout<<"File cant be opened \n";
return 1;          }
fout.write(char *)&savac ,sizeof(customer));
fout.close();
ifstream fin;
fin.open("Saving",ios::in|ios::binary);
fin.read(cahr *)&savac,sizeof(customer));
cout<<savac.name;
cout<<"has the balance Rs"<<savac.balance<<"\n";
fin.close();                }
```

Ans :- Tina Marshall has the balance Rs 21310.75

Q14. Write a program for reading and writing class objects.

```
#include<fstream.h>
#include<conio.h>
class Student
{
char name[40];
char grade;
float marks;
public:
  void getdata(void);
  void display(void);
};
void Student::getdat(void)
{
char ch;
cin.get(ch);
cout<<"Enter name";  cin.getline(name,40);
cout<<"Enter grade";  cin>>grade;
cout<<"Enter marks"; cin>>marks;
cout<<"\n";
}
void Student::display(void)
{
cout<<"Name:"<<name<<"\t"
<<"Grade:"<<grade<<"\t"
<<"Marks:"<<marks<<"\t"<<"\n";
}
void main()
```

```
{
clrscr();
Student arts[3];
fstream filin;
        filin.open("Stu.dat",ios::in|ios::out);
        if(!filin)
        {
        cout<<"Cannot open file \n";
        return 1;
        }
cout<<"Enter details for 3 students \n";
for(int i=0;i<=3;i++)
{
arts[i].getdata();
filin.write(char *) &arts[i],sizeof(arts[i]));
}
filin.seekg(0);
cout<<"the contents of stu.dat are shown below
for(int i=0;i<=3;i++)
{
filin.read(char *) &arts[i],sizeof(arts[i]));
arts[i].display();
}
filin.close();
}
```

Chapter -8

POINTERS

Pointers :

- Pointer is a variable that holds a memory address of another variable.
- It supports dynamic allocation routines.
- It can improve the efficiency of certain routines.

C++ Memory Map :

- Program Code : It holds the compiled code of the program.
- Global Variables : They remain in the memory as long as program continues.
- Stack : It is used for holding return addresses at function calls, arguments passed to the functions, local variables for functions. It also stores the current state of the CPU.
- Heap : It is a region of free memory from which chunks of memory are allocated via DMA functions.

Static Memory Allocation : The amount of memory to be allocated is known in advance and it allocated during compilation, it is referred to as Static Memory Allocation.

Eg. Int a; // This will allocate 2 bytes for a during compilation.

Dynamic Memory Allocation : The amount of memory to be allocated is not known beforehand rather it is required to allocated as and when required during runtime, it is referred to as dynamic memory allocation.

C++ offers two operator for DMA – new and delete.

Free Store : It is a pool of unallocated heap memory given to a program that is used by the program for dynamic memory allocation during execution.

Declaration and Initialization of Pointers :

Datatype *variable_name;

Syntax : Datatype *variable_name;

Int *p; float *p1; char *c;

Eg. Int *p; float *p1; char *c;

Two special unary operator * and & are used with pointers. The & is a unary operator that returns the memory address of its operand.

Eg. Int a = 10; int *p; p = &a;

Pointer arithmetic:

Two arithmetic operations, addition and subtraction, may be performed on pointers.
When you add 1 to a pointer, you are actually adding the size of whatever the pointer is pointing at. That is, each time a pointer is incremented by 1, it points to the memory location of the next element of its base type.
Eg. Int *p; P++;
If current address of p is 1000, then p++ statement will increase p to 1002, not 1001.
If *c is char pointer and *p is integer pointer then

Char pointer	C	c+1	c+2	c+3	c+4	c+5	c+6	c+7
Address	**100**	**101**	**102**	**103**	**104**	**105**	**106**	**107**
Int pointer	p		p+1		p+2		p+3	

Adding 1 to a pointer actually adds the size of pointer's base type.
Base address : A pointer holds the address of the very first byte of the memory location where it is pointing to. The address of the first byte is known as BASE ADDRESS.

Dynamic Allocation Operators :

C++ dynamic allocation routines obtain memory for allocation from the free store, the pool of unallocated heap memory provided to the program. C++ defines two unary operators new and delete that perform the task of allocating and freeing memory during runtime. The operators new and delete are also called as free store operators.

Creating Dynamic Array :
Syntax : pointer-variable = new data-type [size];
Eg. int * array = new int[10];

Not array[0] will refer to the first element of array, array[1] will refer to the second element. No initializes can be specified for arrays.

All array sizes must be supplied when new is used for array creation.
Two dimension array :

Int *arr, r, c;

```
R = 5; c = 5;
Arr = new int [r * c];
Now to read the element of array, you can use the following loops :
For (int i = 0; i < r; i++)
{
        cout << "\n Enter element in row " << i + 1 << " : ";
        For (int j=0; j < c; j++)
                Cin >> arr [ i * c + j];
}
```

Memory released with delete as below :

Syntax for simple variable :	For array :
Delete pointer-variable; Eg. delete p;	delete [size] pointer variable; Eg. delete [ ] arr;

Pointers and Arrays :

C++ treats the name of an array as if it were a pointer i.e. memory address of some element. C++ interprets an array name as the address of its first element.

That is, if marks is an int array to hold 10 integers then marks stores the address of marks[0], the first element of the array i.e., the array name marks is a pointer to an integer which is the first element of array marks[10].

```
void main()
{
        int *m;
        int marks[10];
        cout << "\n Enter marks :";
        for (int i = 0; i < 10; i++)
                cin >> marks[i];
        m = marks;
        cout << "\n m points to " << *m;
        cout << "\n Marks points to " << *marks;
}
```

The name of an array is actually a pointer pointing to the first element of the array.

Since the name of an array is a pointer to its first element, the array+1 gives the address of the second element, array+2 gives the address of the third element, and so on.

Thus, to print the fourth element of array marks, we can give either of the following :
Cout << marks [3]; OR cout << * (marks+3);

Array of Pointers :

To declare an array holding 10 int pointers –

int * ip[10];

That would be allocated for 10 pointers that can point to integers.

Now each of the pointers, the elements of pointer array, may be initialized. To assign the address of an integer variable phy to the forth element of the pointer array, we have to write
ip[3] = & phy;

Now with *ip[3], we can find the value of phy.

int *ip[5];

Index	0	1	2	3	4
address	1000	1002	1004	1006	1008

int a = 12, b = 23, c = 34, d = 45, e = 56;

Variable	a	b	c	d	e
Value	12	23	34	45	56
address	1050	1065	2001	2450	2725

ip[0] = &a; ip[1] = &b; ip[2] = &c; ip[3] = &d; ip[4] = &e;

Index	ip[0]	ip[1]	ip[2]	ip[3]	ip[4]
Array ip value	1050	1065	2001	2450	2725
address	1000	1002	1004	1006	1008

ip is now a pointer pointing to its first element of ip. Thus
ip is equal to address of ip[0], i.e. 1000
*ip (the value of ip[0]) = 1050
* (* ip) = the value of *ip = 12
* * (ip+3) = * * (1006) = * (2450) = 45

Now see the program given below :

```
#include<iostream.h>
Void main()
{
        int x[3][5] = { {1,2,3,4,5}, {6,7,8,9,10}, {11,12,13,14,15} };
        int *n = &x[0][0];
        cout << "\n 1. * ( *(x+2)+1)          =       " << *(*(x+2)+1);
        cout << "\n 2. *(*x+2)+5              =       " <<*(*x+2)+5;
        cout << "\n 3. *(*(x+1))              =       " << *(*(x+1));
```

```
        cout << "\n 4. *(*(x)+2)+1          =     " << *(*(x)+2)+1;
        cout << "\n 5. *(*(x+1)+3)          =     " << *(*(x+1)+3);
        cout << "\n 6. *n                   =     " << *n;
        cout << "\n 7. *(n+2)               =     " << *(n+2);
        cout << "\n 8. (*(n+3)+1)           =     " << (*(n+3)+1);
        cout << "\n 9. *(n+5)+1             =     " << *(n+5)+1;
        cout << "\n 10. ++*n                =     " << ++*n;
}
```

Output with Explanation :

1. * (*(x+2)+1)	=	12	--- *(*(x+2)+1) = *(x[2]+1) = x[2][1] = 12
2. *(*x+2)+5	=	8	--- *(*x+2)+5 = *(x[0]+2)+5 = x[0][2] + 5 = 3+5 = 8
3. *(*(x+1))	=	6	--- *(*(x+1)) = *(x[1]) = *(x[1]+0) = x[1][0] = 6
4. *(*(x)+2)+1	=	4	--- *(*(x)+2)+1 = *(x[0]+2)+1 = (x[0][2])+1 = 3+1 = 4
5. *(*(x+1)+3)	=	9	--- *(*(x+1)+3) = *(x[1] +3) = x[1][3] = 9
6. *n	=	1	--- *n = x[0][0] = 1 (first element)
7. *(n+2)	=	3	--- *(n+2) = x[0][2] = 3 (third element)
8. (*(n+3)+1)	=	5	--- (*(n+3)+1) = x[0][3] (fourth element) + 1 = 4 + 1 = 5
9. *(n+5)+1	=	7	--- *(n+5)+1 = x[1][0] (sixth element) + 1 = 6+1 = 7
10. ++*n	=	2	--- ++ *n = ++ 1 = 2

Pointers and Strings :

Pointer is very useful to handle the character array also.
Eg :

```
Char name[] = "computer";
Char *cp;
For (cp = name; *cp != '\0'; cp++)
        Cout << "--"<<*cp;
```

Output :
--c--o--m--p--u--t--e--r

An array of char pointers is very useful for storing strings in memory.

```
Char *subject[] = { "Chemistry", "Phycics", "Maths", "CS", "English" };
```

In the above given declaration subject[] is an array of char pointers whose element pointers contain base addresses of respective names. That is, the element pointer subject[0] stores the base address of string "Chemistry", the element pointer subject[1] stores the above address of string "Physics" and so forth.

An array of pointers makes more efficient use of available memory by consuming lesser number of bytes to store the string.
An array of pointers makes the manipulation of the strings much easier. One can easily exchange the positions of strings in the array using pointers without actually touching their memory locations.

Pointers and CONST :

A constant pointer means that the pointer in consideration will always point to the same address. Its address can not be modified.

A pointer to a constant refers to a pointer which is pointing to a symbolic constant.

Look the following example :

```
Int m = 20;                          // integer m declaration
Int *p = &m;                         // pointer p to an integer m
++ (*p);                             // ok : increments int pointer p
Int * const c = &n;                  // a const pointer c to an intger n
++ (* c);                            // ok : increments int pointer c i.e. its contents
++ c;                                // wrong : pointer c is const – address can't be modified
Const int cn = 10;                   // a const integer cn
Const int *pc = &cn;                 // a pointer to a const int
++ (* pc);                           // wrong : int * pc is const – contents can't be modified
++ pc;                               // ok : increments pointer pc
Const int * const cc = *k;           // a const pointer to a const integer
++ (* cc);                           // wrong : int *cc is const
++ cc;                               // wrong : pointer cc is const
```

Pointers and Functions :

A function may be invoked in one of two ways :
1. call by value 2. call by reference
The second method call by reference can be used in two ways :
1. by passing the references 2. by passing the pointers

Reference is an alias name for a variable. For ex :

```
Int m = 23;
Int &n = m;
Int *p;
P = &m;
```

Then the value of m i.e. 23 is printed in the following ways :

```
Cout << m;     // using variable name
Cout << n;     // using reference name
Cout << *p;    // using the pointer
```

Invoking Function by Passing the References :

When parameters are passed to the functions by reference, then the formal parameters become references (or aliases) to the actual parameters to the calling function.

That means the called function does not create its own copy of original values, rather, it refers to the original values by different names i.e. their references.

For example the program of swapping two variables with reference method :

```
#include<iostream.h>
Void main()
{
```

```
        void swap(int &, int &);
        int a = 5, b = 6;
        cout << "\n Value of a :" << a << " and b :" << b;
        swap(a, b);
        cout << "\n After swapping value of a :" << a << "and b :" << b;
}
Void swap(int &m, int &n)
{
        int temp;
        temp = m;
        m = n;
        n = temp;
}
```

output :
Value of a : 5 and b : 6
After swapping value of a : 6 and b : 5

Invoking Function by Passing the Pointers :

When the pointers are passed to the function, the addresses of actual arguments in the calling function are copied into formal arguments of the called function.

That means using the formal arguments (the addresses of original values) in the called function, we can make changing the actual arguments of the calling function.

For example the program of swapping two variables with Pointers :

```
#include<iostream.h>
void main()
{
        void swap(int *m, int *n);
        int a = 5, b = 6;
        cout << "\n Value of a :" << a << " and b :" << b;
        swap(&a, &b);
        cout << "\n After swapping value of a :" << a << "and b :" << b;
}
void swap(int *m, int *n)
{
        int temp;
        temp = *m;
        *m = *n;
        *n = temp;
}
```

iutput :
Value of a : 5 and b : 6
After swapping value of a : 6 and b : 5

Function returning Pointers :

The way a function can returns an int, an float, it also returns a pointer. The general form of prototype of a function returning a pointer would be

Type * function-name (argument list);

```
#include <iostream.h>
int *min(int &, int &);
```

```
void main()
{
        int a, b, *c;
        cout << "\nEnter a :";          cin >> a;
        cout << "\nEnter b :";          cint >> b;
        c = min(a, b);
        cout << "\n The minimum no is :" << *c;
}

int *min(int &x, int &y)
{
        if (x < y )
                return (&x);
        else
                return (&y);
}
```

Dynamic structures :

The new operator can be used to create dynamic structures also i.e. the structures for which the memory is dynamically allocated.
struct-pointer = new struct-type;

```
student *stu;
stu = new Student;
```

A dynamic structure can be released using the deallocation operator delete as shown below :
delete stu;

Objects as Function arguments :

Objects are passed to functions in the same way as any other type of variable is passed.
When it is said that objects are passed through the call-by-value, it means that the called function creates a copy of the passed object.
A called function receiving an object as a parameter creates the copy of the object without invoking the constructor. However, when the function terminates, it destroys this copy of the object by invoking its destructor function.
If you want the called function to work with the original object so that there is no need to create and destroy the copy of it, you may pass the reference of the object. Then the called function refers to the original object using its reference or alias.
Also the object pointers are declared by placing in front of a object pointer's name.
Class-name * object-pointer;

Eg. Student *stu;
The member of a class is accessed by the arrow operator (->) in object pointer method.
Eg :

```
        #include<iostream.h>
        class Point
        {
        private :
                int x, y
        public :
                Point()
                {
```

```
        x = y = 0;
    }
    void getPoint(int x1, int y1)
    {
        x = x1; y = y1;
    }
    void putPoint()
    {
        cout << "\n Point : (" << x << ", " << y << ")";
    }
    };
void main()
{
Point p1, *p2;
cout << "\n Set point at 3, 5 with object";
p1.getPoint(3,5);
cout << "\n The point is :";
p1.putPoint();
p2 = &p1;
cout << "\n Print point using object pointer :";
p2->putPoint();
cout << "\n Set point at 6,7 with object pointer";
p2->getPoint(6,7);
cout<< "\n The point is :";
p2->putPoint();
cout << "\n Print point using object :";
p1.getPoint();
}
```

If you make an object pointer point to the first object in an array of objects, incrementing the pointer would make it point to the next object in sequence.

```
student stud[5], *sp;
---
sp = stud;          // sp points to the first element (stud[0])of stud

sp++;               // sp points to the second element (stud[1]) of stud

sp + = 2;           // sp points to the fourth element (stud[3]) of stud

sp--;               // sp points to the third element (stud[2]) of stud
```

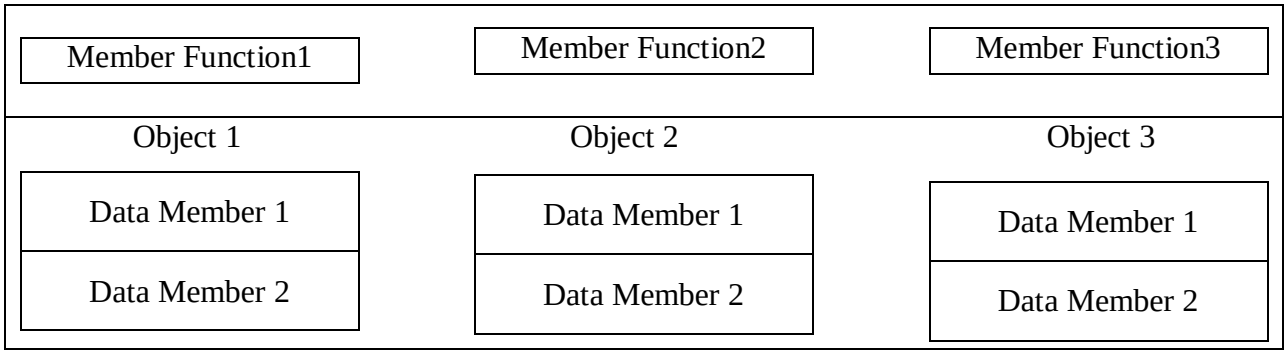
You can even make a pointer point to a data member of an object. Two points should be considered :

1. A Pointer can point to only public members of a class.
2. The data type of the pointer must be the same as that of the data member it points to.

this Pointer :

In class, the member functions are created and placed in the memory space only once. That is only one copy of functions is used by all objects of the class.

Therefore if only one instance of a member function exists, how does it come to know which object's data member is to be manipulated ?



For the above figure, if Member Function2 is capable of changing the value of Data Member3 and we want to change the value of Data Member3 of Object3. How would the Member Function2 come to know which Object's Data Member3 is to be changed ?
To overcome this problem this pointer is used.
When a member function is called, it is automatically passed an implicit argument that is a pointer to the object that invoked the function. This pointer is called This.
That is if ojbect3 is invoking member function2, then an implicit argument is passed to member function2 that points to object3 i.e. this pointer now points to object3.
The friend functions are not members of a class and, therefore, are not passed a this pointer.
The static member functions do not have a this pointer.

Summary :
Pointers provide a powerful way to access data by indirection. Every variable has an address, which can be obtained using the address of operator (&). The address can be stored in a pointer.

Pointers are declared by writing the type of object that they point to, followed by the indirection operator (*) and the name of the pointer. Pointers should be initialized to point to an object or to null (0).
You access the value at the address stored in a pointer by using the indirection operator (*). You can declare const pointers, which can't be reassigned to point to other objects, and pointers to const objects, which can't be used to change the objects to which they point.
To create new objects on the free store, you use the new keyword and assign the address that is returned to a pointer. You free that memory by calling the delete keyword on the pointer. delete frees the memory, but it doesn't destroy the pointer. Therefore, you must reassign the pointer after its memory has been freed.

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;
*i = 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 :

```
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 << " *j = " << *j << ", *k = " << *k << "\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.

(i) 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.

UNSOLVED QUESTIONS

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

```
#include<iostream.h>
int main()
{
```

```
int x[] = { 1, 2, 3, 4, 5 };
for (int i = 0; i < 5; i++)
{
        cout << *x;
        x++;
}
return 0;
}
```

3. Give the output of the following :

```
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 reason for errors.

```
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. Are pointers really faster than array ? How much do function calls slow things down ? Is ++i faster than i = i + 1 ?
7. 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";              }
```

8. Write a function that takes two string arguments and returns a string which is the larger of the two. The larger string has larger ASCII value. Also show how this function will be invoked.
9. 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;                   }
```

10. Give the output of the following program :

```
void main()
{       int array[] = { 2, 3, 4, 5 };
        int *ap = array;
        int value = *ap;
        cout << value << "\n";
        value = *ap++;
        cout << value << "\n";
        value = * ap;
        cout << value << "\n";
        value = * ++ ap;
        cout << value << "\n";          }
```

High Order Thinking Skills (HOTS)

Q1. What is wrong with the following while loops (ans how does the correct ones look like):

(i)
```
int counter =1;
While (counter<100)
{
cout<<counter<<"\n";
counter--;
}
```

(ii)
```
int counter =1;
while (counter <100)
cout<<counter<< "\n";
counter + +;
```

Ans **(i)** In this loop the counter is decremented, so it will have values 1,0,-1,-2,-3…… so this loop is an infinite loop. To fix, we need to use counter ++ instead of counter - - to fix

```
while (counter<100)
{
        ……
        counter + +;
}
```

(ii) In this loop there are not brackets surrounding the code block of the while loop. Therefore, only the line immediately following the while statement repeats. To fix, we need to add grouping bracket around the indented lines after the while statement i.e. as :

```
while (counter<10)
{
cout<<counter<<"\n";
counter++;
}
```

Q2. Write a c++ function that converts a 2-digit octal number into binary number and prints the binary equivalent.

Ans Assume that a header file and main() is including in a program

The function is as follows:

```
Void octobin(int oct)
{
long binn=0;
```

```
int a[6];                /*Each octal digit is converted into 3 bits thus for 2 octal digits
                         -- space for 6 bits has been reserved here*/
int d1,d2,q,r,;
d1=oct%10;
d2= oct/10;
        for int (i=0; i<6; i++)
        {
        a[i] =0;
        }
for (i=0; i<3; i++)
{               q=d1/2;
        r=d1%2;
        a[i]=r;
        d1=q;           }

for ( ; i<6;i++)
{
q=d2/2;
r=d2%2;
a[i]=r;
d2=q;
}
for (i=i-1; i>=0;i - -)
{
binn * =10;
binn += a[i];
}
cout<<endl<<binn<<endl;
}
```

Q3. How we can use arrays as arguments? Explain with example

Ans Array can be used as other data types, as arguments to functions. Here is an example of it

```
// Array as arguments
#include <iostream.h>
const int district=4;
const int months=6;
void display (int [districts][months]);
void main()
{
int d,m;
int sales [districts] [months];
cout<<endl;
for (d=0;d<districts; d++)
for (m=0;m<months; m++)
{
        cout<<"enter sales:"<d++;
        cout<<", months:"<<m+1;
        cin>>sales[d][m];
        }
        display (sales);
        }
```

```
void display (int funcsales [districts] [months])
{
int d,m;
for (d=o; d<districts; d++)
{
cout <<"in district"<< d+1;
for (m=0; m<months; m++)
cout<<funcsales[d][m];
cout<<endl;
}
}
```

Q4. **Write a program that the** roll numbers, marks in English, Computers, Maths out of 100 for 50 students (i.e. need no read them)
Write a function in c++, using structures, to calculate the following:-

(i) No. of students passed with distinction
(ii) Details of top two students
(iii) Number of students failed

For distinction, a student needs to score atleast 75% and minimum marks are 40%

Ans

```
#include<iostream.h>
#include<conio.h>
const int size=50;
struct kvstudent
{
int kvrollno;
float kveng;
float kvcomp;
float kvmaths;
}
kvstudent sarr[size],t1,t2;
float total,avg,top1=0,top2=0;
int ndist=0, nfail=0;
void kvresult()
{
clrscr();
for (int= 0;int<size;i++)
{
total= sarr[i].kveng + sarr[i].kvcomp+sarr[i].kvmaths;
avg= total/3;
if (avg>=75)
ndist++;
else if(avg<40)
nfail++;
if(top1<avg)
{
top1=avg;
t1=sarr[i];
}
```

```
else if (top2<avg && avg <=top1)
{
top2 =avg;
t2= sarr[i];                }         }
cout<< "\n total number of distinction holders are :"<<ndist<< endl;
cout<< "\n toal number of failed students are:"<<nfail<<endl;

cout<< " \n Ist Topper is (Top Scorer) : \n";
cout<< " Roll Number : "<<t1.kvrollno<< "\t English:" <<t1.kveng
        << "\t Computers:" <<t1.kvcomp<<<< "\t Maths:" <<t1.kvmaths
        << "\n Aggregate % :" <<top1<<endl;

cout<< " \n IInd Topper is : \n";
cout<< " Roll Number : "<<t2.kvrollno<< "\t English:" <<t2.kveng
        << "\t Computers:" <<t2.kvcomp<<<< "\t Maths:" <<t2.kvmaths
        << "\n Aggregate % :" <<top2<<endl;
}
```

Q5. What do you think about polymorphism and how you can explain for effective coding as a part of Object Oriented Language?

Ans Polymorphism in an object oriented programming language works under main() and with statements like decision statements and looping statements and works under the pointers and structures. A polymorphism is also be understandable as a function overloading. This is the ability of an object to behave differently in different circumstances can effectively be implemented in programming through function overloading.

It helps in coding to represent the same function in different modes of program. The programmer is relieved from the burden of choosing the right function for a given set of values. This important responsibility is carried out by the compiler when a function is overloaded.

Q6. How we can overload binary operator?. Expalin with example.

Ans **Binary Operator** can be overloaded in the same manner as unary operator. We can take an example of overloading 'equal to' (= =) operator.
We will use this operator to compare the strings, returning values 'true' if strings are same and false otherwise.

Program of overloading binary operator(= =), declaration of functions inside the class.

```
#include <iostream.h>
#include<string.h>
enum boolean { true , false };                    // use of enum
class string
{
private:
char s[100];
public:
string()
{
strcpy(s, " ");
}
```

```
string (char a[])                    // overloading of string function
{
        strcpy (s,a);
}
void display()
{
cout<<s;
}
void gets()
{
cin.get(s,100)
}
Boolean operator = =(string ss)
{
return (strcmp(s,ss.s)= =0)?true :false;
}
};

void main()
{
string s1= "overlaod";
string s2;
cout<< "\n enter a word";
s2.gets();                    //gets strings from user
if (s2= =1)
cout<<" you typed a correct word \n";
else
cout<< " Invalid Match \n";
}
```

Q7. How we can overload constructor?. explain with example.

Ans The constructor is defined as class name. A constructor of a class may also be overloaded so that even with different number and types of initial values, an object may still be initialized.

```
#include<iostream.h>
#include<conio.h>
class Deposit
{
long int principal;
int time;
float rate;
float total_amt;
public:
Deposit();
Deposit(long p, int t, float r);
Deposit(long p, int t);
Deposit(long p, float r);
void calc_amt(void);
void display(void);
};
```

```
Deposit:: Deposit()
{
principal =time=rate=0.0;
}

        Deposit:: Deposit(long p, int t, float r)
        {
        principal=p;
        time=t;
        rate =r;
        }

Deposit:: Deposit(long p, int t)
{
principal=p;
time=t;
rate =0.07;
}

Deposit:: Deposit(long p, float r)
        {
        principal=p;
        time=4;
        rate =r;
        }
void Deposit::calc_amt(void)
{
totat_amt= principal +(principal *time * rate)/100;
}
void Deposit::display (void)
{
cout<< "\n Principal Amount : Rs."<<principal;
cout<< "\n Period of Investment: "<< time << "years";
cout<< "\n Rate of intrest :"<<rate;
cout<< "\n Total amount is:Rs. :- >  " <<total_amt;
}
void main()
{
        clrscr();
        Deposit D1;
        Deposit D2(5000,2,0.05);
        Deposit D3(6000,4);
        Deposit D4(4000,0.08);

        D1.calc_amt();
        D2.clac_amt();
        D3.calc_amt();
        D4.clac_amt();
        cout<< "\n display of object One is: ";
        D1.display();
        cout<< "\n display of object Two is: ";
```

```
D2.display();
cout<< "\n display of object Three is: ";
D3.display();
cout<< "\n display of object Four is: ";
D4.display();
}
```

Q8. Find the errors in the following program. State reasons:

```
#include<iostream.h>

class A
        {
        int  a1;
        public:
        int a2;
        protected:
        int a3;
        };
class B: public A
{
        public:
        void func()
        {
        int b1,b2,b3;
        b1=a1;
        b2=a2;
        b3=a3;
        }
};
class C: A
{
        public:
        void f()
        {
        int c1,c2,c3;
        c1=a1;
        c2=a2;
        c3=a3;
        }
};

        int main()
        {
        int p,q,r,i,j,k;
        B 01;
        C 02;
        p=01.a1;
        q=01.a2;
        r=01.a3;
        i=01.a1;
        j=01.a2;
        k=01.a3;
```

```
return 0;        }
```

Ans The errors in the above given program are as described below:

1. **B:: func()** cannot access **A::a1** as **a1** is a private member of **A, T**herefore, **b1=a1;** **is illegal**
2. **C=f()** cannot access **A::a1** as **a1** is a private number of **A,** Therefore, **c1=a1; is illegal**
3. In **main(),** the statement **p=01.a1;** is illegal because **a1** is not the public member of **B** and hence cannot be accessed directly by its objects.
4. In **main(),** the statement **r = 01.a3; is** illegal for the same reason as specified in point 3.
5. The statements

   i = 02.a1;
   f = 02.a2;
   k = 02.a3;

   are illegal as neither of a1,a2 and a3 are public members of C (which is inheriting privately from A) and hence cannot be accessed directly by the objects of **C** class.

Q9. What will be the output of the following:

```
#include <iostream.h>
void main()
{
        int v1=5, v2=10;
        for (int x=1; x<=2; x++)
        {
                cout<<++v1<< "\t"<<v2--<<endl;
                cout<< --v2<< "\t"<<v1++<<endl;
        }
}
```

Ans The output will be

```
6       10
8       6

8       8
6       8
```

Q10 Write a program that reads a string and counts the number of vowels, words and blank spaces present in the string.

Ans

```
#include<iostream.h>
#include<stdio.h>
main()
```

Q11 Identify the errors in the following code segment

```
int main()
{
cout<< " enter two numbers";
cin>>num>>auto;
float area=length * breadth;
}
```

Ans

1. Variable auto is invalid (it is a reserved keyword)
2. Variables (num,auto (though invalid) are not defined before their usage.
3. Even though variable length and breadth are also not defined
4. Return statement is missisng

Q12 Name the header file for using in built functions in the program
(i) setw(), (ii) puts(), (iii) isdigit(), (iv) fabs()

Ans **Header files are**
(i) iomanip.h, (ii) stdio.h (iii) ctype.h (iv) math.h

Q13. Write a program to generate a function with parameters and array in function e.g. show is function name
then show(int[ ],int);

Ans

```
#include<iostream.h>
#include<conio.h>
main()
{
clrscr();
 int a[5];
 void show(int[],int);                    // declaration of a function (prototype)
 cout<<"enter the number=";
 for(int i=0;i<5;i++)
 {
   cin>>a[i];
 }
 cout<<"ARRAYS"<<endl;
 show(a,5);                               // calling of function
 getch();                          // for freeze the monitor
}
void show(int s[ ],int n)                 //defining of a function
{
 for(int i=0;i<n;i++)
 {
  cout<<s[i]<<endl;
 }
}
```

Q. 14 What will be the output of following code fragment ?

```
#include<iostream.h>
#include<conio.h>
main()
{
clrscr();
int a[] = {3, 5, 6, 7};
int *p, **q, ***r, *s, *t, ** ss;
p = a;
s = p + 1;
q = &s;
t = (*q + 1);
```

```
        ss = &t;
        r = &ss;
        cout << *p << '\t' << **q << '\t' << ***r << end;

    }
```

Ans : 3 5 6

Q15 What is the relationship between an array and a pointer ? Given below a function to tranverse a character array using For-loop. Use a pointer in place of an index X and substitute for-loop with while-loop so that the output of the function stringlength() remains the same.

```
int strlength(char s[])
{
        int count = 0;
        for (int x = 0; s[x]; x++)
                cout ++;
        return (count);          }
```

Ans : The relationship between an array and a pointer is that the name of an array is actually a pointer pointing to the first element of the array.

```
int strlength(char s[])
{
        int count = 0;
        while (*s)
        {
                count++;
                s++;                    }
        return (count);          }
```

Q16 Give the output of the following program segment : (assuming all required header files are included in the program)

```
char *name = "KenDriYa";
for (int x = 0; x < strlen(name); x++)
        if (islower (name[x]} )
                name[x] = toupper (name[x]);
        else
                if ( isupper (name[x]) )
                        if (x%2 != 0)
                                name[x] = tolower (name[x-1])
                        else
                                name[x]--;
cout << name << endl;
```

Ans : jENnRIXA

Q17 What do you under by memory leaks ? What are the possible reasons for it ? How can memory leaks be avoided ?

Ans : If the objects, that are allocated memory dynamically, are not deleted using delete, the memory block remains occupied even at the end of the program. Such memory blocks are known as orphaned memory blocks.

This orphaned memory blocks when increase in number, bring as adverse effect on the system. This situation is known as memory leak. The possible reasons for this are :

(i) A dynamically allocated object not deleted using delete.
(ii) Delete statement is not getting executed because of some logical error.
(iii) Assigning the result of a new statement to an already occupied pointer.

The memory leaks can be avoided by

(ii) making sure that a dynamically allocated object is deleted.
(iii) A new statement stores its return values (a pointer) in a fresh pointer.

Q18 Predict and explain the output of the following program :

Ans :
```
#include<iostream.h>
#include<conio.h>
int main()
{
        clrscr();
        float x = 5.999;
        float *y, *z;
        y = &x;
        z = y;
        cout << x << “, “ << “(&x) << “ , “ << *y << “, “ << *z << “\n”;
        return 0;        }
```

Ans : The output of this program will be

5.999, 5.999, 5.999, 5.999

The reason for this is x gives the value stored in the variable x. *(&x) gives the data value stored in the address &x i.e. address of x i.e. the data value of x. Since y points to x (y = &x), *y gives the value of x. And because z has the same address as that of y. *z also gives the value of x i.e. 5.999.

Q19 Give the output following program :

```
#include<iostream.h>
Int a = 13;
Void main()
{
        Void demo(int &, int , int *);
        Int a = 7, b = 4;
        Demo (::a, a, &b);
        Cout << ::a << “ “ << a << “ “ << b << endl;
}
Void demo(int &x, int y, int *z)
{
        A + = x;
        Y * = a;
        *z = a + y;
        Cout << x << “ “ << y << “ “ << *z << endl;
}
```

Ans :

```
26      182     208
26        7     208
```

Unit – II

Chapter -9 & 10
ARRAYS & LINKED LIST

Defination of Datastructure

A **data structure** is a logical method of representing data in memory using the simple and complex data types provided by the language.

Arrays:

An array is collection of the homogeneous elements that are referred by a common name. It is also called a subscripted variable as the elements of an array are used by the name of an array and an index or subscript.

Array are of two types:

1. One-dimensional arrays 2. Multi-dimensional arrays.

Address Calculation:

The array elements are stored in contiguous memory locations by sequential allocation technique.

Address of arr[i] = B+(I-LB)*S

Sequential Allocation:

The process of storing elements in a fixed order in a data structure where the time required for such access is dependent on the order of the elements.

Using Row Major order the add of a [i] [j] is given by,

Address of a [i] [j]= B+[(i-LB1)*N+(j-LB2)]*S

Using column Major order the add of a [i] [j] is given by,

Address of a [i] [j]= B+[(i-LB1)+(j-LB2)*M]*

One Dimensional Array:

The simplest type of data structure is a one dimensional array in which each elements of a linear array is referenced by one subscript.

The operations one normally perform on any linear structure include the following:

(a) **Traversal**: processing each elements in the list.

(b) **Search**: finding the location of the element with a given value or the record with a given key.

Linear Search: this is a neutral searching method in which we search for a element by traversing through the entire list from beginning until we get the element.

Binary Search: the search can be made faster by an algorithm in which the list or the array should be sorted in advance is called binary search.

(c) **Insertion**: adding a new element to the list.

(d) **Deletion**: removing an element from the list.

(e) **Sorting:** arranging the elements in some types of order.

Insertion sort: this sorting method divides the list into sorted part and unsorted part. It picks up one element from the front of the unsorted part and inserts it at its proper place in the sorted part and repeats this action till the unsorted part is exhausted.

Selection sort: it is very simple way of sorting data stored in a one dimensional array. All the elements in the array are searched for the smallest element. The selected element is exchanged with the first element in the array.

Bubble sort: it is also very simple sorting algorithm. This method proceeds by looking the array for left to right, and whenever a pair of adjacent elements is found to be out of order,

the elements are exchanged. Therefore, after the first pass, the largest element is the array will have bubble up to one end of the array.

(f) **Merging**: merging is the process of combining two or more sorted arrays into another array which is also sorted.

Concatenation of Two Linear Array

Concatenation means joining the element *of two arrays to form a new array.* First copy all the elements of one array and then copy all the elements of other array into the new array. The size of the concatenated array must be equal to or greater than that of the sum of sizes of the two arrays to be concatenated.

Two Dimensional Array:

A two dimensional array(having two subscripts) is suitable for table processing or matrix manipulation.

Traversal: it means visiting each element one after the other.

Finding sum/difference of two N×M arrays containing numeric values:

The sum/difference of two N×M arrays containing numeric values can be obtained by adding/subtracting the corresponding elements of the two arrays. The result can either be directly displayed on monitor or stored into third array of size N×M and then this resulting array can be displayed.

For example, the sum of two arrays A and B of size 3×2 is shown below.

4	6	5	2	9	8
7	1	1	8	8	9
5	3	2	4	7	7
Array A		Array B		resultant after addition	

If difference of two arrays of size 3×3 is required then we get result as illustrated by the following

example :

40	25	73	31	20	41	9	5	32
18	55	29	10	36	15	8	19	14
70	62	47	20	32	17	50	30	30
Array A			Array B			Result after difference		

We may even have negative values as array elements or after sum/difference.

Interchanging Row and Column elements in a Two Dimensional Array:

Interchanging means swapping the values. The elements of row and column can be interchanged only if the array is of size N×N as the result is to be stored in the array itself.

For example,

1	2	3	1	4	7
4	5	6	2	5	8
7	8	9	3	6	9
Given Array			Array after interchange		

Stacks :

A stack is a list of elements in which an element may be inserted or deleted only at one end, called the TOP of the stack. This means, in particular, that elements are removed from a stacks in the reverse order of that in which they were inserted into the stack.

A stack is a **last-in first-out** or ***LIFO*** data structure.

(a) "Push" s the term used to insert an element into a stack.
(b) "Pop" is the term used to delete an element from the stack.

Queue:

A queue is a linear list of elements in which deletions can take place only at one end, called the front, and insertions can take place only at the other end, called the rear. The terms "front" and "rear" are used in describing a linear list only when it is implemented as a queue.
A queues are also called **first-in first-out(FIFO)** list.

Linked List:

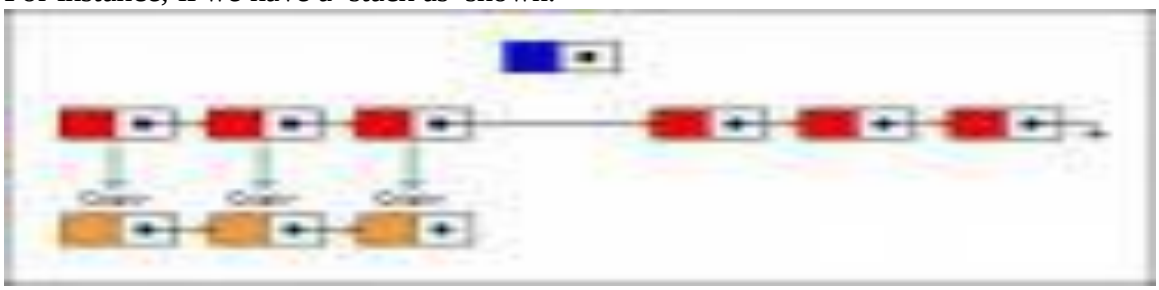
A linked list is a linear collection of data elements, called nodes pointing to the next nodes by means of pointers.

Stack As A linked List

A linked list is a dynamic data structure where space requirements need not be predetermined. A stack implemented as a linked list also inherits all these properties. The creation of a stack (as a linked list) is the same as the creation of a linked list I.E, after getting a node for the ITEM to be inserted, TOP(pointer pointing to the top) points to the newly inserted node.

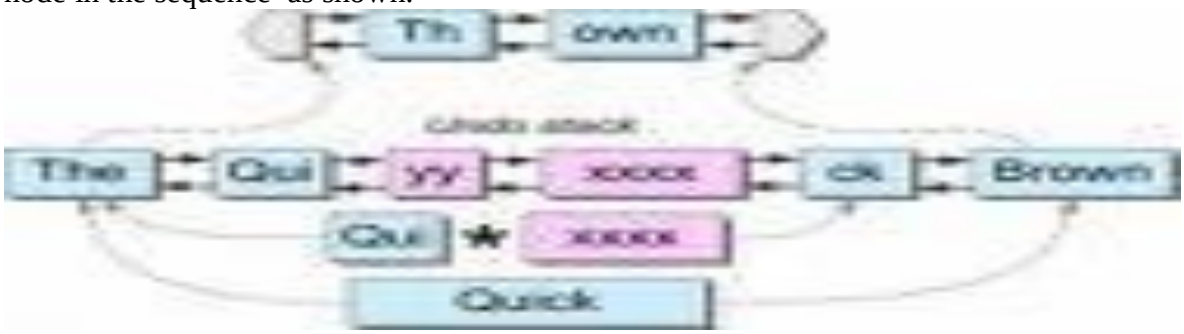
Insertion In Alinked Stack (Pushing):--

As a push can occur only at the top , top gets modified every time.
For instance, if we have a stack as shown:---



Deletion from a linked stack (popping) :--

Deletion i.e., popping also requires modification of top , that is , top is made to point to the next node in the sequence as shown:---



Solved Questions

Q1. What is the difference between linear and non- linear data structures?

Ans. Single level data structures where elements form a sequence are called linear data structures eg. Stacks, queues, linked list etc. are linear data structures.

Multilevel data structures are called non-liner data structures eg. Trees and graphs are non-linear data structures.

Q2. Describe the similarities and differences between queues and stacks.

Ans. Similarities:

(a) Both queues and stacks are special cases of linear lists.

(b) Both can be implemented as arrays or linked lists.

Dissimilarities :

(a) A Stack is a LIFO list, a queue is a FIFO list.

(b) There are no variations of a stack, a queue, however, may be circular or dequeue.

Q3. What is meant by the term "Overflow" & "Underflow"?

Ans. "Overflow" means attempt to INSERT when the list is full and no free space is available. "Underflow" means attempt to DELETE an item from an empty list.

Q4 . Distinguish between infix, prefix and postfix algebric expression giving examples of each.

Ans. Infix Notation : In this notaion, the operator symbol is placed in between the operands eg.

A+B, (A-C)* B

Prefix Notation : In this notation, the operator symbol is placed before its operands eg.

+AB, *-ACB

Postfix Notation : In this notation, the operator symbol is placed after its operands eg.

AB+, AC-B*, ABC*+

Q5 . Evaluate the following postfix notation of expression, show status of stack for each operation

500, 20, 30, + , 10, * , +

Ans. Adding] to mark the end of given postfix expression ie. 500, 20, 30 +, 10, *, +]
First 500 will be added to the stack, then 20 and then 30.
Then we get an operator + so 20 will be added to 30 and pushed to the stack. Then 10 is pushed to the stack.
Then * operator is found thus it multiplies 10 with 50 and the result is pushed into the stack.
Then again + operator is found which adds 500 with 500 that is placed in the stack. And we get the result 1000.

Unsolved Questions

Q. 1. How is computer memory allotted for a 2D array?

Q. 2. Binary search is to be used on the following sorted array to search for 30 and 60.

***Index*:**	1	2	3	4	5	6	7	8	9	10
***Value*:**	11	22	30	33	40	44	55	60	66	70

Give the index of the element that would be compared with at every step. Repeat the process replacing 30 by 60.

Q. 3. Consider the single dimensional array AAA [45] having base address 300 and 4 bytes is the size of each element of the array. Find the address of AAA [10], AAA [25] and AAA [40].

Q. 4. Given two dimensional array A[10][20], base address of A being 100 and width of each element is 4 bytes, find the location of A[8][15] when the array is stored as *a) column wise b) Row wise.*

Q. 5. Write a C++ function to find and display the sum of each row and each column of a 2 dimensional array of type float. Use the array and its size as parameters with float as the return type.

Q. 6. Differentiate between a FIFO list and LIFO list.

Q. 7. Transform the following expression to prefix and postfix form:
(A+B*C-D)/E*F

High Order Thinking Skills (HOTS)

Q1. What are preconditions for Binary search to be performed on a single dimensional array?

Ans. For binary search
(a) the list must be sorted
(b) lower bound and upper bound of the list must be known

Q2. How is computer memory allotted for two dimensional array?

Ans. For two-dimensional array, computer memory is allocated either in Row-major form or column-major form.
Row-major form stores the 2-D array row wise ie. firstly the first row is stored, then the second row, then the third row and so forth.
Column-major form stores the 2-D array columnwise ie. firstly the first column, then the second column, then the third column and so on.
The default form is Row-major form.

Q3. Calculate the address of X[4,3] in a two dimensional arrayX[1….5,1….4] stored in a row major order in the main memory. Assume the base address to be 1000 and that each element requires 4 words of storage.

Ans.

Base address		B=1000
Word size		W=4 bytes
No. of columns		n=4
First row number		L1=1
First column number		L2=1
Using the formula		
Address		X[I,T] = B + W[n(I-L1)+(J+L2)]
Given I=4, J=3		
Address of	X[4,3]	=1000+4[4(4-1)+(3-1)]
		=1000+4(4(3)+2)
		=1000+4(14)
		=1000+56 =1056

Q. 4. Transform the following expressions to infix form:
1. + - ABC
2. + A – BC

Q. 5. Evaluate the following postfix expression using a stack and show the contents of the stack after execution of each operation
5, 6, 9, +, 80, 5, *, -, /

Q 6. Give the necessary declaration of a linked implemented stack containing integer type numbers; also write a user defined function in C++ to pop a number from this stack.

UNIT –III
Chapter 11 & 12
DATABASE CONCEPTS & SQL

Database :
Database is a collection of different kinds of data which are connecting with some relation.
In different way, it is a computer based record keeping system. The collection of data referred to as a database.

Purpose of Database :
Years ago, for keeping record, a typical file-processing system is used. A number of different application programs are written to extract records from and add records to the appropriate files. But this scheme has a number of major limitations and disadvantages, such as data redundancy, data inconsistency, unsharable data, unstandardized data, insecure data, incorrect data, etc.

A database management system is answer to all these problems as it provides a centralized control of the data.

1. Databases reduce the data redundancy to a large extent
2. Databases can control data inconsistency to a large extent
3. Databases facilitate sharing of data
4. Databases enforce standards
5. Databases can ensure data security
6. Integrity can be maintained through databases

Data : It is row facts, figures, characters, etc with which we have to start.
Information : The processed data or meaningful data is known as information.
Database : An organized collection of relational data is called as database.
DBMS : It is a Database Management System. It is a system to used for create database, store the data, secure it and fetch data whenever required. MS Access, Oracle is the example of DBMS.

Database Abstraction :
A major purpose of a database system is to provide the users only that much information that is required by them. This means that the system does not disclose all the details of data, rather it hides certain details of how the data is stored and maintained. A good database system ensures easy, smooth and efficient data structures in such a way so that every type of database user : end user, application system analyst, and physical storage system analyst, is able to access its desired information efficiently.

Various Levels of Database Implementation :

1. **Internal Level (Physical Level) :** This level describes how the data are actually stored on the storage medium. At this level, complex low-level data structures are described in details. Eg. RollNo is in Byte(4), offset = 34.
2. **Conceptual Level :** It describes what data are actually stored in the database. It also describes the relationships existing among data. Eg. ItemCode is in Numeric(5).
3. **External Level (View Level – Logical Level) :** This is the level closest to the users. It is concerned with the way in which the data are viewed by individual users. Most of the users of the database are not concerned with all the information contained in the database. Eg. View 1 contains RollNo, Name, DOB.

Data Independence :

The ability to modify a scheme definition in one level without affecting a scheme definition in the next higher level is called Data Independence.

i. **Physical Data Independence :** It refers to the ability to modify the scheme followed at the physical level without affecting the scheme followed at the conceptual level.

ii. **Logical Data Independence :** It refers to the ability to modify the conceptual scheme without causing any changes in the schemes followed at view levels.

Data Models :

The external level and conceptual level user certain data structures to help utilize the database efficiently. There are three data models that are used for database management system are :

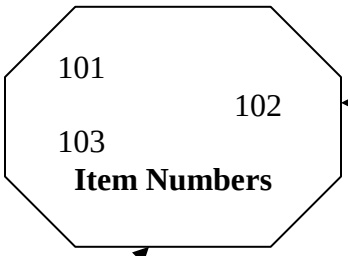
1. Relational Data Model 2. Hierarchical Data Model 3. Network Data Model

THE RELATIONAL MODEL :

The relational model was propounded by E.F. Codd of the IBM and has since been acknowledged as a very important concept in DBMS. The relational model has established itself as the primary data model for commercial data processing applications.

Relation : A relation is a table, i.e. data arranged in rows and columns.

Domain : It is a pool of values from which the actual values appearing in a given column are drawn.



Item#	Item_Name	Fund
101	Computer	VVN
102	Chair	SF
103	Table	SF

Item_Master

Purchase_Master

Item#	Pur_Date	Supplier	Quantity
101	25/04/2007	Super Mart Computers	20
101	15/05/2008	Supreme Computers	25
102	21/03/2007	Sharma Furnitures	50
102	11/12/2007	Shiv Furnitures	24
103	12/02/2008	Sharma Furnitures	30

Eg. The values appearing in the ItemCode# column of both the Item_Master table and the Pur_Master are drawn from the underlined domain of all valid Item Codes.

A domain is said to be atomic if elements of the domain are considered to be indivisible units.

Tuple : The rows of tables are generally referred to as Tuples.

Attributes : The columns of tables are generally referred to as attributes.

Degree : The number of attributes in a relation determine the degree of a relation. A relation having 5 attributes is said to be a relation of degree 5.

Cardinality : The number of tuples in relation is called the Cardinality of the relation. Eg. Cardinality of Purchase_Master relation is 5.

Views :

A view is a kind of table whose contents are taken from other tables depending upon a conditions. Views do not contain data of their own. The contents of a view are determined by carrying out the execution of the given query.

A view is a virtual table that does not really exists in its own right but is instead derived from one or more underlying base table(s).

Keys : It is important to be able to specify how rows in a relation are distinguished conceptually, rows are distinct from one another, but from a database perspective the difference among them must be expressed in terms of their attributes. Keys come here for a rescue.

Primary Key : It is a set of one or more attributes that can uniquely identify tuples within the relation. Eg. Item# is the primary key for Item_Master.

Candidate Key : All attributes combinations inside a relation that can serve as primary key are Candiadate keys as they are candidates for the Primary Key position.

Alternate Key : A candidate key that is not the primary key is called an alternate key.

Foreign Key : A non-key attribute, whose values are derived from the primary key of some other table, is known as Foreign Key in its current table. Here Item# of Purchase_Master is foreign key, because it is derived from Item_Master.

Referential Integrity : It is a system of rules that a DBMS uses to ensure that relationships between records in related tables are valid, and that users don't accidentally delete or change related data.

The Relational Algebra :

It is set of operations that can be used to manipulate relations. Each operation operates on one or more than one relation and produces a new relation.

(i) **Union :** If the degree of two relations is same and their corresponding attributes are defined on the same domain then the relations are called union compatible. The operator for union is '$\cup$';
Ex : C = A $\cup$ B

(ii) **Intersection :** If two relations A and B are union compatible then they can also participate in an intersection operation. Its result is the common tuples. The operator for intersection is '$\cap$';
Ex : C = A $\cap$ B

(iii) **Difference :** The operator is '-';
Ex : C = A – B
The tuples of C are belonging to A which do not belong to B.

(iv) **Cartesian Product** : It returns a concatenation of each tuple of A with each tuple of B.
Ex : C = A X B

(v) **Select :**Whenever user desires to search or select tuples of a relation depending upon a condition being true, the operation is known as a select operation. The operator is 'σ'.
Ex : $\sigma_{marks > 95}$ (Students)

(vi) **Projection :** A projection operation on a relation produces a vertical subset of the relation in the sense that it returns the relation minus certain columns.
Ex : $\pi_{name, roll}$ (Students)

(vii) Natural Join (|X|) :
If two relations A and B have at least one common attribute then they can participate in a Join operation. The resultant relation contains the attributes of both A and B.

SQL – Structured Query Language :

SQL is the set of commands that is recognized by nearly all RDBMS. It is a language that enables us to create and operate on relational databases, which are sets of related information stored in tables.

Processing Capabilities of SQL :

1. **Data Definition Language :** The SQL DDL provides commands for defining relation schemas, deleting relations, creating indexes, and modifying relation schemas.
2. **Interactive Data Manipulation Language :** The SQL DML includes a query language based on both the relational algebra and the tuple relational calculus. It includes also commands to insert, delete and modify tuples in the database.
3. **Emedded Data Manipulation Language :** The embedded form of SQL ids designed for use within general-purpose programming languages such as PL/1, Cobol, Fortran, etc.
4. **View Definition** : The SQL DDL also includes commands for defining views.
5. **Authorization** : The SQL DDL includes commands for specifying access rights to relations and views.
6. **Integrity :** The SQL provides forms of integrity checking. Future products and standards of SQL are likely to include enhanced features for integrity checking.
7. **Transaction Control :** SQL includes commands for specifying the beginning and ending of transactions along with commands to have a control over transaction processing.

Data Definition Language :
A database scheme is specified by a set of definitions which are expressed by a special language called a data definition language (DDL). The result of compilation of DDL statements is a set of tables which are stored in a special file called data dictionary or directory.

A Data Dictionary is a file that contains "Metadata" i.e. "Data about Data".

Classification of SQL Statements :

1. Data Definition Language Commands
2. Data Manipulation Language Commands
3. Transaction Control Language Commands
4. Session Control Commands
5. System Control Commands

DDL Commands :
It allows us to perform tasked related to data definition. Through these commands user can perform tasks like :

i. Create, alter and drop schema objects :
CREATE TABLE, ALTER TABLE, DROP TABLE, CREATE INDEX, ALTER INDEX, DROP INDEX, RENAME, TRUNCATE, etc.

ii. Grant and revoke privileges and roles :
These commands are also known as Data Control

iii. Analyse, audit or add command :
ANALYSE, AUDIT, NOAUDIT, COMMENT

DML Commands :
It is a language that enables users to access or manipulate data as organized by the appropriate data model.

The DMLs are basically of two types :

1. Procedural DMLs require a user to specify what data is needed and how to get it.
2. Non-procedural DMLs require a user to specify what data is needed without specifying how to get it.

INSERT INTO, DELETE, SELECT, LOCK TABLE, etc.

TCL Commands :

A transaction is successfully completed if and only if all its constraints steps are successfully completed. To manage and control the transactions, the transaction control commands are used.
COMMIT, ROLLBACK, SAVEPOINT, SET TRANSACTION

Schema Objects :

A schema refers to the collection of logical structures of data.

Eg. Clusters, Database triggers, Indexes, Packages, Stored Functions, Procedures, Tables, Views.

Data Types :

Data Types are means to identify the types of data and associated operations for handling it.

There are various data types in Oracle :

1. Varchar2(size) : Variable length character string having maximum length size bytes.
2. Number(p, s) : Number having precision p and scale s.
3. Long : Character data of variable length up to 2 gigabytes.
4. Date : Valid date range from January 1, 4712 BC to December 31, 4712 AD.
5. Raw(size) : Raw binary data of length size bytes.
6. Long Raw : Raw binary data of variable length up to 2 gigabytes.
7. Rowid : Hexadecimal string representing the unique address of a row in its table.
8. char(size) : Fixed length character data of length size bytes.

Solved SQL Command With Syntax

1. Create Table : To create a table.

Syntax :

CREATE TABLE <table name>
(<column name> <data types>[(size)] [,<column name> <data types>[(size)]….);

Eg.

```
Create table student
(rollno number(2),
name varchar2(20),
dob date);
```

Constraints :

Constraints are the means by which user can prevent invalid data entry into the table.

i. **Not Null constraint :** It ensures that the column cannot contain a NULL value.

```
Create Table Student
(RollNo number(2) Not Null,
Name varchar2(20) Not Null,
Dob date,
City varchar2(20) );
```

ii. **Unique constraint :** A candidate key is a combination of one or more columns, the value of which uniquely identifies each row of a table.

```
Create table student
(RollNo number(2) Unique,
Name varchar2(20) Not null,
Dob date,
City varchar2(20) );
```

iii. **Primary Key** : It ensures two things : (i) Unique identification of each row in the table. (ii) No column that is part of the Primary Key constraint can contain a NULL value.

```
Create table student
(RollNo number(2) Primary Key,
Name varchar2(20) Not null,
Dob date,
City varchar2(20) );
```

iv. **Foreign Key** : The foreign key designates a column or combination of columns as a foreign key and establishes its relationship with a primary key in different table.

```
Create table Fee
(RollNo number(2)  references Student (Rollno),
Name varchar2(20) Not null,
Amount number(4),
Fee_Date date);
```

v. **Check Constraint :** Sometimes we may require that values in some of the columns of our table are to be within a certain range or they must satisfy certain conditions.

```
Create table Employee
(EmpNo number(4) Primary Key,
Name varchar2(20) Not null,
Salary number(6,2) check (salary > 0),
DeptNo number(3)
);
```

2. Create View :

A view is defined as a logical table based on a database table or another view.

Syntax :

Create View <view name> As Select <column list> from <table name>[Where <condition>];

Eg. Create View Stud_Jaipur As Select * from Student Where City = 'Jaipur';

3. Drop Table :

It is used to delete a table.

Syntax : Drop Table <table name>;

Eg. Drop Table student;

4. Drop View :

It is used to delete a view.

Syntax : Drop View <view name>;
Eg. Drop view stud_Jaipur;

5. Alter Table : To change or modify the definition of table.

(i) To add column :
Alter table Student
Add Class number(2);

(ii) To modify a column definition :
Alter table Student
Modify (rollno number(3));

(iii) To drop a column :
Alter table student
Drop column city;

6. Update --- Set --- :
It is used to modify the data of a column in a table.
Syntax :
Update <table name>
Set <column name> = <value> [,<column name> = <value>, ….]
[where <condition>];

Eg.
Update student
Set name = 'Rahul Sharma'
Where rollno = 5;

7. Insert into :
To insert data into the table.
Syntax :
Insert into <table name> [<column list>]
Values (<list of values>);

1. Omitting column list in Insert Clause :
Insert into Students
Values(1, 'Amit Sharma', '12-Jun-1990', 'Jaipur');

2. Listing columns in the Insert Clause :
Insert into Students (rollno, name)
Values (23, 'Abhishek Mehta');

8. Delete :
To delete values from the table.

1. Delete from Students;
It deletes all the records from the table students.
2. Delete from students where city = 'Ajmer';
It deletes all the records from the table students where city is Ajmer.

9. Select :
To fetch data from the table or a group of tables.

Syntax :

SELECT [distinct] <attribute name> [, <attribute name>] …
FROM <relation list>
[WHERE <condition>];
Examples :

(i) select * from students;
- It lists all data from the table students.

(ii) select rollno, name from students;
- It displays only rollno and name of the table students.

(iii) select * from students where class = 12;
- It displays data from table students for class 12 only.

(iv) select * from students where class = 12 and city = 'Jaipur';
- It displays data whose class is 12 and city is Jaipur.

(v) select name from students where name like 'ABH%';
- It displays name of students which is started with ABH.

(vi) select distinct city from students;
- It displays the distinct value of column city from the table students.

(vii) select rollno, name from students order by dob;
- It displays rollno and name from the table students in order of their dob.

(viii) select rollno, name from students order by dob desc;
- It displays rollno and name from the table students in descending order of their dob.

SQL Functions :

SQL supports functions which can be used to compute and select numeric, character and date columns of a relations. These functions can be applied on a group of rows. The rows are grouped on a common value of a column in the table. These functions return only one value for a group and therefore, they are called aggregate or group functions.

1. **SUM() :**
   It returns the sum of values of numeric type of a column.
   Eg. Select sum(salary) from employee;
2. **AVG() :**
   It returns the average of values of numeric type of a column.
   Eg. Select avg(salary) from employee;
3. **MIN() :**
   It returns the minimum of the values of a column of a given relation.
   Eg. Select min(salary) from employee;
4. **MAX() :**
   It returns the maximum of the values of a column of a given relation.
   Eg. Select max(salary) from employee;
5. **COUNT():**
   It returns the number of rows in a relation.
   Eg. Select count(*) from employee;

Group By Clause :

The rows of a table can be grouped together based on a common value by using the Group By clause of SQL in a select statement.
Syntax :
SELECT <attribute name>, <attribute name> ---- [functions]
FROM <relation name>

GROUP BY <group by column>;
Eg.
Select age, count (rollno)
From students
Group by age;

Output :

Age	Count(rollno)
15	2
14.5	2
14	5

Group-By-Having Clause :

It is used to apply some condition to the Group By clause.
Eg.
Select class
From students
Group by class
Having count(*) > 5;

Unsolved Questions :

1. Write SQL commands for (i) to (iv) and give outputs for (v) to (viii) for the given table Library.

B_Code	Title	Author	Type	Pub	Price
102	C++	S P Mathur	Prog	McGraw	150
206	VB	M K Sharma	Prog	BPB	175
355	DOS Guide	Norton	OS	PHI	250
392	Mastering Windows	Cowart	OS	BPB	510
455	Network Concepts	Norton	Net	ZPress	410

(i) Select all the BPB publication books from the library.
(ii) Display different book type from the library.
(iii) Display book title in order by their price.
(iv) Insert a record in table with the details (324, JAVA, 'Palmer', 'Prog', 'McGraw', 300)
(v) Select count(*) from library;
(vi) Select max(price) from library where pub = 'BPB';
(vii) Select sum(price) from library where type = 'OS';
(viii) Select title, type from library order by B_Code desc;

2. **Write SQL commands for (i) to (vi) for the given table Employee.**

EmpNo	Name	Designation	Dept	Salary
2045	M K Sharma	Clerk	Finance	15000
1265	S P Yadav	Manager	Finance	25000
4502	M L Mourya	Account	Finance	20000
4007	J P Singh	Clerk	Computer	12500
1101	K Natarajan	Administrator	Purchase	32500

(i) Display all the records of the table Employee

(ii) Display name and designation of employee order by name
(iii) Display different department.
(iv) Display all employees whose department is Finance and salary is more than 20000.
(v) Find out the maximum salary of the employee.
(vi) Display all employees whose name starts with 'M'.

High Order Thinking Skills (HOTS)

Q1. (a) Differentiate between the terms primary key and alternate key.

Ans. All candidate keys, which are not the primary key of the table are called alternate keys.

Q2. Consider the following tables Consignor, Consignee and Consignment. Write SQL commands for the statements (i) to (iv) and give the outputs for SQL queries (v) to (viii).

Sender

SenderID	SenderName	SenderAddress	City
ND01	R Jain	2, ABC Appts	New Delhi
MU02	H Sinha	12, Newtown	Mumbai
MU15	S Jha	27/A, Park Street	Mumbai
ND50	T Prasad	122 – K, SDA	New Delhi

Recipient

RecID	SenderID	RecName	RecAddress	RecCity
KO05	ND01	R Bajpayee	5, Central Avenue	Kolkata
ND08	MU02	S Mahajan	116, A Vihar	New Delhi
MU19	ND01	H Singh	2A, Andheri East	Mumbai
MU32	MU15	P K Swamy	B5, C S Terminus	Mumbai
ND48	ND50	S Tripathi	13, B I D, Mayur Vihar	New Delhi

(i) To display the names of all Senders from Mumbai

Ans. SELECT sendername from Sender where sendercity='Mumbai';

(ii) To display the RecIC, Sendername, SenderAddress, RecName, RecAddress for every Recipient.

Ans. Select R.RecIC, S.Sendername, S.SenderAddress, R.RecName,

R.RecAddress from Sender S, Recepient R

where S.SenderID=R.SenderID;

(iii) To display Recipient details in ascending order of RecName

Ans. SELECT * from Recipent ORDER By RecName;

(iv) To display number of Recipients from each city

Ans. SELECT COUNT(*) from Recipient Group By RecCity;

(v) SELECT DISTINCT SenderCity from Sender;

Ans. SenderCity

Mumbai

New Delhi

(vi) **SELECT A.SenderName, B.RecName**

From Sender A, Recipient B

Where A.SenderID = B.SenderID AND B.RecCity =’Mumbai’;

Ans.

A.SenderName	**B.RecName**
R Jain	H Singh
S Jha	P K Swamy

(vii) **SELECT RecName, RecAddress**

From Recipient

Where RecCity NOT IN (‘Mumbai’, ‘Kolkata’);

Ans.

RecName	RecAddress
S Mahajan	116, A Vihar
S Tripathi	13, BID, Mayur Vihar

(viii) **SELECT RecID, RecName**

FROM Recipent

Where SenderID=’MU02’ or SenderID=’ND50’;

Ans.

RecID	RecName
ND08	S Mahajan
ND48	S Tripathi

Unit -IV
CHAPTER- 13
BOOLEAN ALGEBRA

Boolean algebra

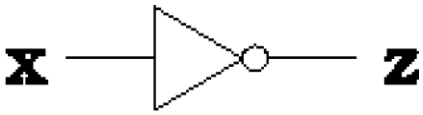
Boolean algebra:- a modern algebra which uses the set of numbers 0 and 1,the logic numbers used to solved logic problems.
Binary decision:-. The decision which results into yes or no. Also called logical statements or truth function.
Truth table:- a table representing all possible input-output combinations for a given logical problem/expression.
Tautology:-a Boolean expression that always results in true or 1.
Fallacy:-a Boolean expression that always results in false or 0.
Cononical expression :- a Boolean expression having all minterms or maxterms.
Minterm:-product of all the literals(with or without the bar) within the logic system.
Maxterm:-sum of all the literals(with or without the bar) within the logic system.
Karnaugh Map:- it is a graphical representation of the truth table of the given expression.

Logic Gates

Logic gates serve as the building blocks to digital logic circuits using combinational logic. We're going to consider the following gates: NOT gates (also called inverters), AND gates, OR gates, NAND gates, NOR gates, XOR gates, and XNOR gates.
We'll also discuss the concept of gate deltay.

NOT gates

NOT gates or *inverters* have a single bit input and a single bit of output.
This is a diagram of a NOT gate. It is a triangle with a circle on the right. The circle indicates "negation".



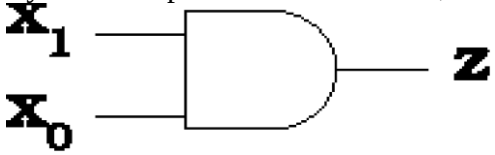
The truth table defines the behavior of this gate.

x	z
0	1
1	0

where **x** is the input and **z** is the output.

AND_2 gates

AND_2 gates have two bits of input and a single bit of output. The subscript, 2, indicates how many inputs this AND gate has. For example, AND_3 gates have 3 inputs.
The output of AND_2 gate is 1 only if both inputs are 1. Otherwise, the output is 0.



The truth table defines the behavior of this gate.

x_1	x_0	z
0	0	0
0	1	0
1	0	0
1	1	1

The function implmented by **AND_2** gates has interesting properties:

1. The function is symmetric. Thus, **x * y == y * x**. This can be verified by using truth tables. We use ***** to represent AND_2.
2. The function is associative. Thus, **(x * y) * z == x * (y * z)**. This can be verified by using truth tables.

Because of these properties, it's easy to define **AND_n**, which is an n-input **AND** gate.

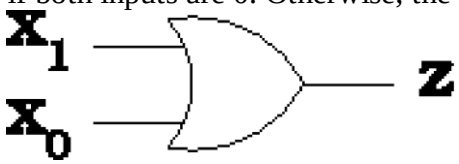
$$AND_n(x_1, x_2,...,x_n) = x_1 * x_2 * ... * x_n$$

That is, an AND gate with n-inputs is the AND of all the bits. This is not ambiguous because the AND function is associative (all parenthesization of this expression are equivalent).

OR_2 gates

OR_2 gates have two bits of input and a single bit of output. The subscript, 2, indicates how many inputs this OR gate has. For example, OR_3 gates have 3 inputs.

The output of OR_2 gate is 0 only if both inputs are 0. Otherwise, the output is 1.



The truth table defines the behavior of this gate.

x_1	x_0	z
0	0	0
0	1	1
1	0	1
1	1	1

The function implemented by **OR_2** gates has interesting properties:

1. The function is symmetric. Thus, **x + y == y + x**. This can be verified by using truth tables. We use "+" to represent **OR_2**
2. The function is associative. Thus, **(x + y) + z == x + (y + z)**. This can be verified by using truth tables.

Because of these properties, it's easy to define **OR_n**, which is an n-input **OR** gate.

$$OR_n(x_1, x_2,...,x_n) = x_1 + x_2 + ... + x_n$$

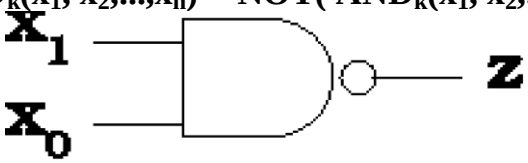
That is, an AND gate with n-inputs is the AND of all the bits. This is not ambiguous because the AND function is associative (all parenthesization of this expression are equivalent).

$NAND_2$ gates

$NAND_2$ gates have two bits of input and a single bit of output. The subscript, 2, indicates how many inputs this NAND gate has. For example, $NAND_3$ gates have 3 inputs.

$NAND_k$ gates is define unusually. Since $NAND_2$ is *not* associative, the definition is based on AND_2. In particular

$$\mathbf{NAND_k(x_1, x_2,...,x_n) = NOT(AND_k(x_1, x_2,...,x_n))}$$



Thus, $NAND_k$ is the negation of AND_k.

The truth table defines the behavior of this gate. It's the negation of **AND_2**.

x_1	x_0	z
0	0	1
0	1	1
1	0	1
1	1	0

The function implemented by **$NAND_2$** gates has interesting properties:

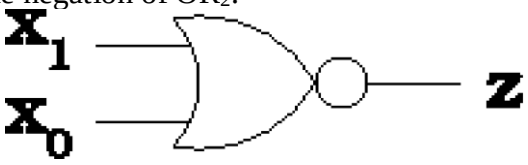
1. The function is symmetric. Thus, **x NAND y == y NAND x**. This can be verified by using truth tables.
2. The function is *not* associative. This can be verified by using truth tables.

Because of these properties, **$NAND_k$** is defined from **AND_k**, and *not* built from **$NAND_2$** gates.

NOR_2 gates

OR_2 gates have two bits of input and a single bit of output. The subscript, 2, indicates how many inputs this OR gate has. For example, NOR_3 gates have 3 inputs.

The output of NOR_2 gate is the negation of OR_2.



The truth table defines the behavior of this gate.

x_1	x_0	z
0	0	1
0	1	0
1	0	0
1	1	0

The function implmented by **NOR_2** gates has interesting properties:

1. The function is symmetric. Thus, **x NOR y == y NOR x**. This can be verified by using truth

tables.

2. The function is *not* associative. This can be verified by using truth tables.

Because of these properties, **NOR_k** is defined from **OR_k**, and *not* built from **NOR_2** gates.

XOR_2 gates

XOR_2 gates have two bits of input and a single bit of output.

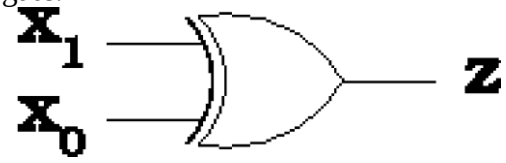
The output of XOR_2 gate is 1 only if the inputs have opposite values. That is, when one input has value 0, and the other has value 1.. Otherwise, the output is 0.

This is called *exclusive-or*. The definition of OR_2 is *inclusive-or*, where the output is 1 if either input is 1, or if both inputs are 1.

XOR_2 can be defined using AND_2, OR_2, and NOT.

x XOR y == (x AND (NOT y)) OR ((NOT x) AND y) == x\y + y\x

Here's a diagram of the XOR_2 gate.



If you look carefully at the drawing of the gate, there is a second arc behind the first one near the inputs. Since this second arc is hard to see, it's usually a good idea to write the word "XOR" inside the gate.

The truth table defines the behavior of this gate.

x_1	x_0	z
0	0	0
0	1	1
1	0	1
1	1	0

The function implmented by **XOR_2** gates has interesting properties:

1. The function is symmetric. Thus, **x (+) y == y (+) x**. This can be verified by using truth tables. (We use (+) to denote logical XOR--ideally, we'd draw it with a + sign inside a circle, but HTML doesn't seem to have a symbol for this).
2. The function is associative. Thus, **[x (+) y] (+) z == x (+) [y (+) z]**. This can be verified by using truth tables.

Because of these properties, it's easy to define **XOR_n**, which is an n-input **XOR** gate.

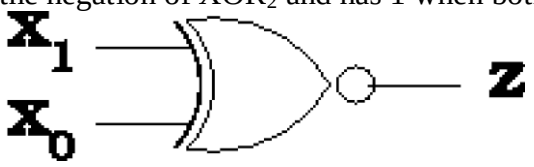
$$XOR_n(x_1, x_2,...,x_n) = x_1 (+) x_2 (+) ... (+) x_n$$

That is, an XOR gate with n-inputs is the XOR of all the bits. This is not ambiguous because the XOR function is associative (all parenthesization of this expression are equivalent).

$XNOR_2$ gates

$XNOR_2$ gates have two bits of input and a single bit of output.

The output of $XNOR_2$ gate is the negation of XOR_2 and has 1 when both inputs are the same.



If you look carefully at the drawing of the gate, there is a second arc behind the first one near the inputs. Since this second arc is hard to see, it's usually a good idea to write the word "XNOR" inside the gate.
The truth table defines the behavior of this gate.

x_1	x_0	z
0	0	0
0	1	1
1	0	1
1	1	0

The function implmented by **$XNOR_2$** gates has interesting properties:

1. The function is symmetric. Thus, **x XNOR y == y XNOR x**. This can be verified by using truth tables.
2. The function is associative. Thus, **(x XNOR y) XNOR z == x XNOR (y XNOR z)**. This can be verified by using truth tables.

Because of these properties, it's easy to define **$XNOR_n$**, which is an n-input **XNOR** gate.

$$XNOR_n(x_1, x_2,...,x_n) = x_1 \text{ XNOR } x_2 \text{ XNOR } ... \text{ XNOR } x_n$$

That is, an XNOR gate with n-inputs is the XNOR of all the bits. This is not ambiguous because the XNOR function is associative (all parenthesization of this expression are equivalent).
(Error-checkers! You may wish to verify this, and email me if this is incorrect!).

Solved questions

Q1. State the principal of duality in Boolean algebra.

Ans Principal of duality states that from every Boolean relation, another Boolean relation can be derived by

i) Changing each OR sign(+) to an AND sign(.)
ii) Changing each an AND sign(.) to an OR sign(+)
iii) Replacing each 1 with 0 and each 0 with 1

Q2. Define the following terms:

(a) Logical constant
(b) Logical variable
(c) Binary valued quantity
(d) Boolean literal

Ans:- (a) the truth values true(1) or false(0)are known as logical constants.
(b) A variable that can store the truth-values (TRUE or FALSE) is called logical variable.
(c) quantity can be represented in terms of TRUE or FALSE is known as binary valued quantity.
(d) a single Boolean variable(logical variable) or its complement e.g X or Y or $\bar{Z}$ is known as literals.

Q3. State Demorgan's laws.

Ans Demorgan's first law: -it states that $\overline{X+Y} = \bar{X}.\bar{Y}$

Demorgan's second law: -it states that $\bar{X}.\bar{Y} = \overline{X+Y}$

Q4 . Why are AND and NOR gates called Universal Gates?

Ans. Nand and Nor gates are less expensive and easier to design.also,other switching function(And ,OR) can easily be implemented using NAND/NOR gates.thus,these (NAND,NOR) Gates are also referred to as universal gates

Q5. Given the following truth table, derive a sum of product (SOP) and product of sum (POS) form of Boolean expression from it:

X	Y	Z	G(X,Y,Z)
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

Ans. In SOP $F= \sum(1,2,3,5,7)$

$= \bar{X}\ \bar{Y}Z+ \bar{X}\ Y\ \bar{Z}+ X\ \bar{Y}Z+XYZ$

In POS $F=\pi(0,3,4,6)$

$=(X+Y+Z)(X+\ \bar{Y}+\ \bar{Z})\ (\ \bar{X}+Y+Z)(\ \bar{X}+\ \bar{Y}+Z).$

Q6. Prove that X.(X+Y)=X by truth table method.

Ans.

X	Y	X+Y	X.(X+Y)
0	0	0	0
0	1	1	0
1	0	1	1
1	1	1	1

From the above table it is obvious that X.(X+Y) = X because both the columns are identical.

Q7. Prove that X.(X+Y)=X by algebric method

Ans. LHS = X.(X+Y)

= X.X+X.Y

=X+X.Y

=X.(1+Y)

=X.1 = X = RHS

Q8. State the distributive laws of Boolean algebra. How are they different from distributive laws of ordinary algebra.

Ans. Distributive laws of Boolean algebra state that

(i) X(Y+Z) = XY+XZ

(ii) X+YZ =(X+Y)(X+Z)

Ist law X(Y+Z) = XY+XZ holds good for all values of X, Y and Z in ordinary algebra whereas X+YZ =(X+Y)(X+Z) holds good only for two values (0,1) of X, Y and Z

Q9. In Boolean algebra, verify using truth table that (X + Y)' = X' Y' for each X, Y in (0, 1).

Ans. As it is a 2-variable expression, truth table will be as follows :

X	Y	X+Y	(X+Y)'	X'	Y'	X'Y'
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

State Demorgan's laws. Verify one of the Demorgan's laws using truth tables.

Ans. De Morgan's first theorem. It states that $X + Y = X \cdot Y$

De Morgan's second theorem. It states that $X \cdot Y = X + Y$

Truth table for second theorem

X	Y	X.Y	X.Y	X	Y	X+Y
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

X.Y and X+Y are identical.

Q10. Why are AND and NOR gates called Universal gates?

Ans. NAND and NOR gates are less expensive and easier to design. Also, other switching functions (AND, OR) can easily be implemented using NAND/NOR gates. Thus, these (NAND, NOR) gates are also reffered to as Universal Gates.

Q11. By means of truth table, demonstrate the validity of the following Postulates / Laws of Boolean algebra:

(a) Commulative law
(b) Absorption law
(c) Idempotent law

Ans The commulative law states that

(i) X+Y = Y+X (ii) X.Y =Y.X

(i) Truth table for X+Y = Y+X is given below :

Input		**Output**	
X	**Y**	**X+Y**	**Y+X**
0	0	0	0
0	1	1	1
1	0	1	1
1	1	1	1

Comparing the columns X+Y and Y+X, we see both of these are identical. Hence proved.

(ii) Truth table for X.Y = Y.X is given below :

Input		Output	
X	**Y**	**X.Y**	**Y.X**
0	0	0	0
0	1	0	0
1	0	0	0
1	1	1	1

Comparing the columns X.Y and Y.X, we see both of these are identical. Hence proved.

The absorption law states that

(i) X+XY = X (ii) X(X+Y) =X

(i) Truth table for X+XY = X is given below :

Input		Output	
X	**Y**	**XY**	**X+XY**
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

Both the columns X+XY and X are identical. Hence proved.

(ii) Truth table for X.(X+Y) = X is given below :

Input		Output	
X	**Y**	**X+Y**	**X(X+Y)**
0	0	0	0
0	1	1	0
1	0	1	1
1	1	1	1

Column X and X(X+Y) are identical. Hence proved.

The Idempotent law states that

(i) X+X = X ii) X.X =X

(i) Truth table for X+X = X is given below :

Input		Output
X	**X**	**X+X**
0	0	0
1	1	1

(ii) Truth table for X.X = X is given below :

Input	Output

X	X	X.X
0 1	0 1	0 1

Q12. Obtain the simplified form of a boolean expression using Karnaugh map.

$F(u,v,w,x) = \sum (0, 3, 4, 5, 7, 11, 13, 15)$

[00]WZ	[01] WZ	[11]WZ	[10]WZ
1		1	
1	1	1	
	1	1	
		1	

2 quads, 1 pair.

Quad 1(m3+m7+m11+m15) reduces to WZ

Quad 2(m5+m7+m13+m15) reduces to VZ

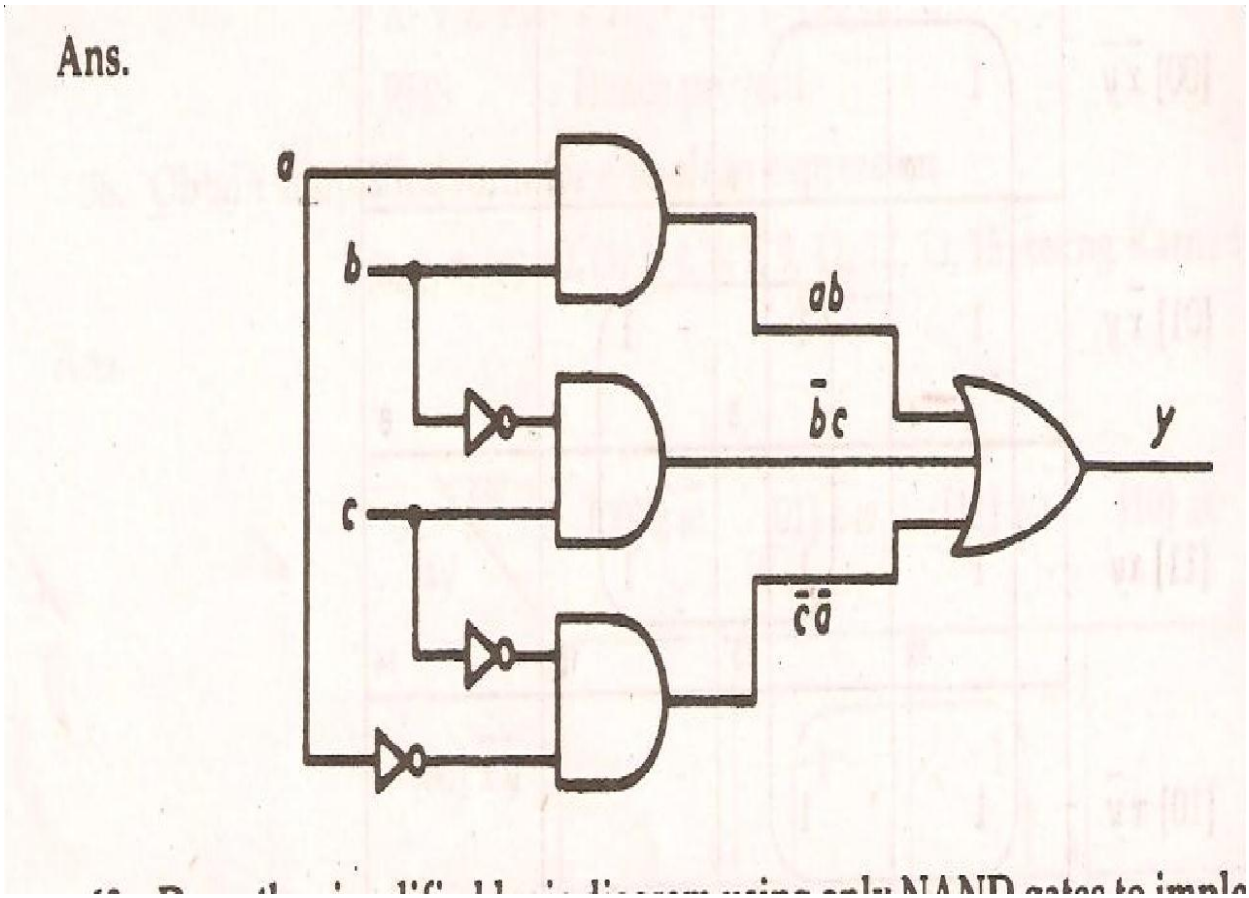
Pair 1(m0,m4) reduces to UWZ

Therefore F=WZ + VZ + UWZ

Q13 . Draw the logic circuit diagram for the following expression :

Y = a b + b c + c a

Ans.

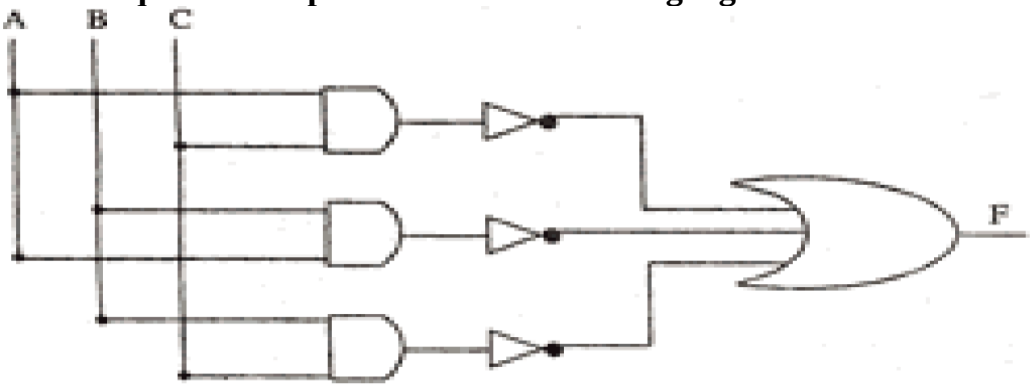


Q14. Prepare a truth table for X Y Z + X Y

Ans. Truth table for X Y Z + X Y is given below :

Input			Output					
X	**Y**	**Z**	**X'**	**Y'**	**Z'**	**X'YZ'**	**XY'**	**X'YZ' + XY'**
0	0	0	1	1	1	1	0	0
0	0	1	1	1	0	0	0	0
0	1	0	1	0	1	1	0	1
0	1	1	1	0	0	0	0	0
1	0	0	0	1	1	1	1	1
1	0	1	0	1	0	0	1	1
1	1	0	0	0	1	1	0	0
1	1	1	0	0	0	0	0	0

Q15. Write the equivalent expression for the following logic circuit :



Ans. F=(AC)' +(BA)'+(BC)'

Q16. Draw the circuit diagram for F = AB'C + C'B using NAND to NAND logic only.

Ans. F=AB'C + C'B =((A)NAND(B')NAND(C))NAND((C')NAND B)

Q17. Write the Sum of Products form of the function G(U,V,W). Truth table representation of G is as follows :

U	V	W	G
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Ans. To get the product of sums form, we need to add maxterms for all those input combinations that produce output as 0. Thus ,

G(U,V,W) = (U + V + W) (U + V + W') (U + V' + W') (U' + V + W')

Ans. (X + Y')(X' + Y)(X' + Y')

Unsolved Questions:

1. Define Binary logic.
2. What is a Boolean Operation ?
3. Define a boolean function.
4. Define a Boolean Expression.
5. Name the three primary and secondary operators of Boolean Algebra.
6. State any four postulates of boolean algebra.
7. Define Idempotent Law & Absorptive Law.
8. Define Involution Law.
9. What is De Morgan's Theorem ?
10. State the principle of duality.
11. Define the Sum Of Products format of a boolean expression.
12. Define the Product of Sums format of a boolean expression.
13. What is a Full Adder ?
14. Differentiate between an Encoder and a Decoder ?
15. Define the working of a XOR gate ?
16. What is a Canonical Sum of Products ?
17. What is a Canonical Product of Sums ?
18. State and verify duality principle.
19. What do you understand by a minterm and a maxterm?
20. F(x,y,z,w)=∑(1,3,4,5,7,9,11,12,13,15) using k-map.

High Order Thinking Skills(HOTS)

Q1. **State the principle of duality in Boolean Algebra and give the dual of the Boolean ex-pression (X + Y) . (Xf + Zf) . (Y + Z)**

Ans: Principle of Duality states that from every Boolean relation, another boolean relation can be derived by Changing each OR sign to AND and vice versa Replacing each 1 by 0 and vice versa. The new derived relation is known as the dual of the original relation.
Dual of (X + Y) . (X□f + Z□f) . (Y + Z) is X.Y + X□f.Z□f + Y.Z

Q2. **Sev**en inverters are cascaded one after another. What is the output if the input is 1?

Ans: 0.

Q3. **Why are NAND and NOR gates called Universal Gates?**

Ans: NAND and NOR gates are less expensive and easier to design. Also other functions (NOT, AND, OR) can easily be implemented using NAND/NOR gates.

Q4. **Obtain a simplified form for the following Boolean ex-pression using Karnaugh□fs Map:**
F(a, b, c, d) = ‡"(0, 1, 2, 4, 5, 7, 8, 9, 10, 11, 14).

Ans:

1	1	0	1
1	1	1	0
0	0	0	1
1	1	1	1

Quad 1 = afcf , Quad 2 = abf , Pair 1 is afbd, Pair 2 is bfcdf , Pair 3 is acdf
The simplified form is: afcf + abf + afbd + bfcdf + acdf

Q5. Prove XY+YZ+YZ□f=Y algebraically.

Ans:- XY+YZ+YZ□f
=XY+Y(Z+Z□f)
=XY+Y=Y(1+X)=Y hence proved

Q6. Simplified ABfCDf+ABfCD+ABCDf+ABCD.

Ans. ABfCDf+ABfCD+ABCDf+ABCD
= ABfC(Df+D)+ABC(Df+D)
= ABfC.1+ABC.1
= AC(Bf+B)
= AC.1=AC.

Q7. Draw the diagram of digital circuit for F(a,b,c)= AB+BC+CD using NAND-to- NAND logic.

Ans. F(a,b,c)= AB+BC+CD
=(A NAND B) NAND (B NAND C) NAND (C NAND D)
Thus the logic circuit is

Q8. Prepare a truth table for XfYf+XfY

Ans XfYf+XfY is a 2- Variable ex-pression ,its truth table is as follows

X	Y	Xf	Yf	XfYf	XfY	XfYf+XfY
0	0	1	1	1	0	1
0	1	1	0	0	1	1
1	0	0	1	0	0	0
1	1	0	0	0	0	0

Q9. Convert X+Y into minterms.

Ans X+Y=X.1+Y.1
=X(Y+Y□f)+Y(X+X□f)
=XY+XY□f+XY+X□fY
=XY+XY+XY□f+X□fY
=XY+XY□f+X□fY.

Q10. Convert the following function into canonical product of sums form F(X,Y,Z)=Π(0,2,4,5).

Ans $F(X,Y,Z)=\Pi(0,2,4,5)= M_O.M_2.M_4.M_5$
M0=000=X+Y+Z
M2=010=X+Y□f+Z
M4=100=X□f+Y+Z
M5=101=X□f+Y+Z□f
F= (X+Y+Z)(X+Y□f+Z)(X□f+Y+Z)(X□f+Y+Z□f)

Unit – V
CHAPTER-14
Communication and Network Concpets

Key Points

- A **network** is a collection of interlinked computers by means of a communication system.
- **Switching techniques** are used for transmitting data across networks.
- A network can have any of these transmission media or connecting media: twisted pair cable, coaxial cable, optical fibre, microwave , radio wave , satellite etc.
- The pattern of interconnection of nodes in a network is called topology.
- Most popular topologies are star , bus , ring , graph and mesh, tree
- RJ 45 (Registered Jack 45) is a eight wire connector, which is commonly used to connect computers on LANs – especially Ethernets.
- Protocol:- A protocol means the rules that are applicable for a network. Protocol defines standardized formats for data packets, techniques for detecting and correcting errors and so on.
- Some most common protocols are HTTP ,FTP, TCP/IP
- **Wireless communication** is simply data communications without the use of landlines.
- **TDMA**:-Time Division Multiple Access technology divides a radio frequency into time slots and then allocates allots to multiple calls.
- **CDMA**:-Code division Multiple access uses a spread spectrum technique where data is sent in small pieces over a number of discrete frequencies available for use. Each user's signal is spread over the entire bandwidth by unique spreading code. At the receivers end, the same unique code is used to recover the signal.
- **SMS** is the transmission of short text message to and from a mobile phone, fax machine and/or IP address.
- **Email (Electronic Mail**) is sending and receiving message by computer.
- A **web Browser is a WWW** client that navigates through the World Wide Web and displays web pages .A web server is a WWW server that responds to the request made by web browser.
- **A URL(**Uniform Resource Locator) specifies the distinct address for each resource on the internet. An Internet address, which is character, based is called domain name.
- **Cookies are messages** that a Web server transmits to a web browser so that the web server can keep track of the user's activity on a specific Web site.

Solved Questions

Q1. What factors affect data transmission?

Ans Several factors affect how data is transmitted. They include the following

(i) Transmission rate-Frequency and bandwidth
(ii) Line Configurations-Point to point versus multipoint.
(iii) Serial and Parallel transmission
(iv) Direction of transmission-Simplex, Half Duplex, and Full Duplex
(v) Transmission Mode-Asynchronous and synchronous.
(vi) Circuit switching and Packet switching
(vii) Multiplexing (viii)Protocols

Q2. What is a Network? What are the benefits of networks?

Ans A network or communications network, is a system of interconnected computers, telephones or other communications devices that can communicate with one another and share applications and data.

The benefit of networks are given below:

(i) Sharing of peripheral devices.
(ii) Sharing of programs and data
(iii) Better communications
(iv) Security of information
(v) Access to databases

Q3. Define the following terms:
(i) Node (ii) Hub (iii) Backbone(iv) Router (v) Bridge (vi) Gateway

Ans (i) **Node**:- A node is a piece of hardware on the system that can be addressed by a message from another node, that is , a computer, printer , fax, modem or CD-ROM drive.

(ii) **Hub**:- Node are connected to a hub, also called a concentrator, whose purpose is to simplify wiring of nodes to each other and to route signals between the nodes.

(iii) **Backbone**:-A backbone is a high –capacity link to which many nodes or hub can be connected ,it is design to carry lots of traffic.

(iv) **Router**:-A router is a special computer that direct communicating messages when several networks are connected together.

(v) **Bridge:-**A bridge is an interface used to connect the same type of networks.

(vi)**Gateway**:- A gateway is an interface permitting communication between dissimilar networks-for instance ,between a LAN and a WAN or between two LANs based on different network operating system or different layouts.

Q4. What are the major difference between LAN and WAN?

Ans The major differences between LAN and WAN

LAN	WAN
• LAN are small sized networks • Range of LAN may vary from 10m to 1.5 km. • LAN operate between 10Mbps and 2 Gbps • They offer very low delay • Complete ownership by a single organization • Very low error rates	• WAN are geographically spread over wide area. • There is no limitation on the distance it spans. • Typical speed of WAN varies from 56Kbps to 155 Mbps. • They offer very high delay • Owned by multiple organizations • Comparatively higher error rates

Q5. Give two major reasons to have network security.

Ans Two major reasons to have network security are given below:

(i) Secrecy:- Keeping information out of the reach of unauthorized users.
(ii) Authentication:- Determining the authorized users before sharing sensitive information with or entering into a business contract.

Q6. What is the purpose of using a Web Browser? Name any one commonly used Web Browser.

Ans The Web Browser fetches the page requested, interprets the text and formatting commands that it contains and display the page property formatted on the screen.

Q7. Write two advantages and disadvantages for STAR topology.

Ans **Advantages of STAR topology:**

(i) One device per connection.

(ii) Easy to access

Disadvantages of STAR topology:

(i) Central node dependency

(ii) Long cable length.

Q8. Write one difference between Telnet and FTP.

Ans Telnet-to connect to remote computers. Telnet is a program or command that allows the user to connect to remote computers on the Internet using a user name and password.

FTP(File transfer protocol) is a method whereby the user can connect to a remote computer known as FTP site and transfer files to his/her own microcomputer's hard disk. Some FTP files are open to the public, some are not

Q9. Explain the following terms in short:

(i) DHTML (ii) ISP

Ans **DHTML** is the next generation of HTML.It describes how text and images are displayed on a web page. Dynamic HTML, developed by Netscape and the World Wide Web Consortium (W3C) is based entirely on industry standard HTML and Java. New features in Dynamic HTML, such as absolute positioning and layers , give designers and developers much greater control over the look and feel of web pages.

ISP(Internet Service Provider) is a company that connects your communication line to is servers, or central (host)computer, which connects you to the internet via another company's network access points. ISP's have a wide range of prices and packages for users to choose from

Q10. What are firewalls?

Ans Firewalls are defensive barriers that fence off a private network from the internet.

Q11. What is the difference between Message Switching technique and Packet Switching technique?

Ans Message Switching:-It is better known as store and forward. In this mechanism a node on receiving a message, stores it till the appropriate route is not free, then forwards it on the route when the route is free.

Packet Switching:-It is best for data. In a packet –switched network, data are transmitted in discrete units called packets. A packet is a fixed length block of data for transmission. The maximum length of the packet is established by the network. Longer transmissions are broken up into multiple packets. The packets have headers with priority codes and source and destination addresses along with data to be sent. The packets are sent over the network node to node, kept there for a small time and then routed according to the information in its header

Q12. Write two applications of Cyber –Law

Ans Two applications of Cyber –Law are given below

(i) Restricting/Penalising unathorised user

(ii) Promoting , coordinating and controlling e-Business.

Q13. Write a not on Fast Ethernet Technology.

Ans The growing importance of LANs and the increasing complexity of desktop computing applications are fueling the need for high performance networks. 100BASE-T(Fast Ethernet) provides a cost effective and high-performance for small workgroups. SMBs(Small to Medium Business), and any network supporting bandwidth-intensive applications. Fast Ethernet technology operates at 10 times the speed of traditional Ethernet, offering maximum performance and enhanced capability for existing Ethernet- based networks.

Q14. What is the diffrence between XML and HTML? Write two differences.

Ans The major differences between XML and HTML

XML	HTML
Does'nt specify either semantics or tag set. It is a language for documents containing structured information and all semantics are defined by the applications that process them.	The semantics and tag set are fixed. It is a language used to design the layout of a document and to specify the hyperlinks.

Q15. Expand the following terminologies:
(i) TCP/IP (ii) XML (iii) CDMA (iv) WLL

Ans (i) TCP/IP:-Transmission Control Protocol/Internet Protocol
(ii)XML:-eXtensible Markup Language
(iii)CDMA:-Code division Multiple access
(iv) WLL:-Wireless Local Loop.

Q16. What is a Modem?What is its function?

Ans A modem is a computer peripheral that connects a worstation to other workstation via telephones lines and facilitates communications.It is short form for modulation/Demodulation.
Modem converts digital signals to A/F(Audio Frequncy) tones which are in the frequency range that the telephone lines can transmit and aloso it can convert transmitted tones back to digital information.

Q17. Differentiate between Internet and Intranet.

Ans The internet is world wide network of computer networks around the globe.Internet uses a set of protocols called TCP/IP .
On the other hand Intranet is a network ,whichi is privately owned. Intranet also uses a set of protocols as internet.

Q18. Define the following :
(i) Data channel (ii) Baud (iii)bps (iv) Bps (v) Bandwidth

Ans

(i) **A data channel** is a medium used to carry information or data from one point to the other.

(ii) **Baud** is the unit of measurment for the information carrying capacity of a communication channel.It is synonymous with bits per second.

(iii) **bps**:- bits per second.It refers to a thousand **bits** transmitted per second

(iv) **Bps**:- bytes per second.It refers to a thousand **bytes** transmitted per second

(v) Bandwidth:-It refers to the diffrence between the highest and lowest fequencies of a transmission channel.This term is also sometimes used to refer to the amount of information travelling through a single channel at any one point of time.

Q 19) Differentiate between tree and Bus topologies of a network.

A19) **Tree topology** is a network with the shape of an inverted tree with the central root branching and sub-branching to the extremities of the network.
Bus topology:-In this topology , all devices on network are connected to a single continous cable called a bus.This topolgy can be used for for smaller networks.

Unsolved Questions

Q1) What do you mean by network topology ? What are the most popular topologies.

Q2) What are repeaters and routers?

Q3) What are protocols

Q4) Differentiate between GSM and CDMA

Q5) Write a short note on WLL.

Q6) Differentiate between downloading and uploading.

Q7) Give one advantage and one disadvantage of optical fibre cable and coaxial cable used in communication.

High Order Thinking Skills(HOTS)

Q1. What do you mean by a backbone network?

Ans. A backbone network is a network that is used as a backbone to connect the connect several LAN'S together to form a WAN.
FDDI (Fiber Distribute Data Interface) is such a network. FDDI is a high performance fiber optic token ring LAN running at 100 Mbps over distances up to 200 kms with up to 1000 stations connected.

Q2. What are cookies?

Ans. Cookies are messages that a web server transmits to a web browser so that the web server can keep track of the user's activity on a specific web site.

Q3. What is XML? How is it different from HTML?

Ans. XML is a markup language for creating documents containing structured information.
In HTML both tag semantics and tag are fixed but XML specifies neither semantics nor tagset.Rather it provides a facility to define tags and relationships among them.

Q4. What are switched LANs/Switched Ethernet LANS?

Ans. LANs that are segmented through switches are called switched LANs. In case of Ethernet LANs they are called Switched Ethernet LANS.

Q5. What is a communication channel? What choices do u have while choosing a communication channel for a network?

Ans. Communication channels mean the connecting cables that link various workstations. There are three basic types of cables:

- **(i) Twisted Pair Cables**:- These cables consist of two insulated copper wires twisted around each other. These are also used for short and medium range telephone communication.
- **(ii) Coaxial Cables**:- A coaxial cable consist of one or more small cables in protective covering. These are more expensive than twisted pair cable but perform better.
- **(iii) Fiber-optic Cables:**- These cables are made of plastic and glass and are about as thick as human hair. These cables are highly durable and offer excellent performance but are expensive.

Q6 What do you mean by topology? What are the most popular topologies?

Ans. Topology refers to the way which the workstations attached to the network are interconnected. The most popular topologies are:

Bus, Ring ,Star,Tree

Q7. What is firewall? What are different firewall techniques?

Ans. The system designed to prevent unauthorized access to or form a private network is called FIREWALL.

There are several types of firewall techniques:
Packet Filter, Application Gateway, Circuit-level Gateway, Proxy Server

Q8. What is the significance of anonymous user in FTP? What is uploading? Can an anonymous user upload? If yes, how? If no, why?

Ans. In order to use FTP effectively, one needs to be an authorized user. However, anonymous FTP is a method whereby FTP server allows the general public to access files on the FTP. Uploading of files refers to the transfer of files from one's computer onto FTP server. An anonymous user generally is not allowed to do so and anonymous users have permission to download only.
However, on many UNIX FTP servers, there is a / incoming directory. If this directory is there, then an anonymous users have permissions to upload only in this directory and no where else.

Q9. What is Ethernet? What is Ethernet Card?

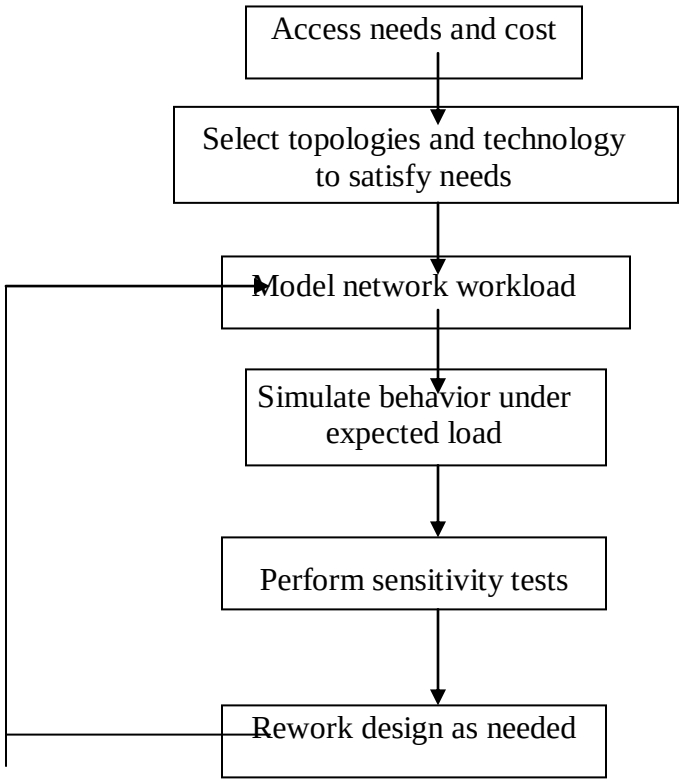
Ans. Ethernet is a LAN architecture developed by Xerox Corp in association with DEC and Intel. Ethernet uses bus or star topologies and can support data transfer rates of up to 10 Mbps.
The omputers that are the part of Ethernet have to install a special card called Ethernet Card.

Q10. What is 80-20 rule of network design?

Ans. The 80-20 rule of network says that : 80% of the traffic on a given network segment should be local and not more than 20% of the network traffic should need to move across a backbone i.e. the spine connecting various sub networks.

Q11. What is the general process of designing networks?

Ans. The general process of designing networks is described through following flow chart:



Model Test Paper – 1

Time allowed : 3 hours *Maximum Marks : 70*

Instructions :

(i) ***All*** *questions are compulsory.*

(ii) Programming Language : C++

1. (a) Differentiate between a Call by Value ‘and Call by Reference, giving suitable examples of each. 2

(b) Name the header files to which the following belong: 1

(i) abs( )

(ii) strcmp( )

(c) Rewrite the following program after removing the syntactical error(s), if any. Underline each correction. 2

```
#include <iostream.h>
const int Multiple 3;
void main ()
{
        Value=15;
        for (int Counter = 1;Counter=<5;Counter++,Value-=2)
                if (Value%Multiple==0)
                        cout<<Value * Multiple;
                        cout<<endl;
                else
                        cout<<Value+Multiple<<endl;
}
```

(d) Find the output of the following program: 3

```
#include<iostream.h>
struct MyBox
{
        int Length, Breadth, Height;
};
void Dimension (MyBox M)
{
        cout<<M.Length<<"x"<<M.Breadth<<"x";
        cout<<M.Height<<endl;
}
void main ()
{
        MyBox B1={10,15,5}, B2, B3;
        ++B1.Height;
        Dimension(B1);
        B3 = B1;
        ++B3.Length;
        B3.Breadth++;
        Dimension(B3);
        B2 = B3;
        B2.Height+=5;
        B2.Length--;
        Dimension(B2);
}
```

(e) Find the output of the following program: 2

```
#include<iostream.h>
#include<string.h>
#include<ctype.h>
void Convert(char Str[],int Len)
{
        for (int Count =0; Count<Len; Count++ )
        {
                if (isupper (Str [Count] ) )
                Str[Count]= tolower(Str[Count]);
                else if (islower (Str [Count] ) )
                Str[Count]= toupper(Str[Count]);
                else if (isdigit (Str [Count]))
                Str[Count]=. Str[Count] + 1i
                else Str[Count] = '*';
        }
}
void main ()
{
        char Text [] = "CBSE Exam 2005";
        int Size=strlen(Text);
        Convert(Text,Size);
        Cout<<Text<<endl;
        for (int C = 0,R=Size-l;C<=Size/2; C++,R--)
        {
                char Temp = Text[C];
                Text [C] = Text [R] ;
                Text [R] = Temp;
        }
        cout<<Text<<endl;
}
```

(f) Observe the following program SCORE.CPP carefully, if the value of Num entered by the user is 5, choose the correct possible output(s) from the options from (i) to (iv), and justify your option. 2

```
//program : SCORE.CPP
#include<stdlib.h>
#include<iostream.h>
void main()
{
        randomize ();
        int Num, Rndnum;
        cin>>Num;
        Rndnum = random (Num) + 5;
        for (int N = 1; N<=Rndnum; N++)
        cout<<N<<" ";
}
```

Output Options:
(i) 1 2 3 4
(ii) 1 2
(iii) 1 2 3 4 5 6 7 8 9

(iv) 1 2 3

2. (a) Define the term Data Hiding in the context of Object Oriented Programming. Give a suitable example using a C++ code to illustrate the same. 2

(b) Answer the questions (i) and (ii) after going through the following class: 2

```
class Test
{
        char Paper[20];
        int Marks;
        public:
        Test () // Function 1
        {
                strcpy (Paper, "Computer")
                Marks = 0;
        }
        Test (char P [] ) // Function 2
        {
                strcpy(Paper,P);
                Marks = 0;
        }
        Test (int M) // Function 3
        {
                strcpy(Paper,"Computer");
                Marks = M;
        }
        Test (char P[], int M) // Function 4
        {
                strcpy (Paper, P);
                Marks = M;
        }
};
```

(i) Which feature of Object Oriented Programming is demonstrated using Function 1, Function 2, Function 3 and Function 4 in the above class Test?

(ii) Write statements in C++ that would execute Function 2 and Function 4 of class Test.

(c) Define a class TravelPlan in C++ with the following descriptions : 4

Private Members:

- PlanCode of type long
- Place of type character array (string)
- Number_of_travellers of type integer
- Number_of_buses of type integer

Public Members:

A constructor to assign initial values of Plan Code as 1001, Place as "Agra", Number_of_travellers as 5, Number_of_buses as 1

A function NewPlan( ) which allows user to enter PlanCode, Place and Number_of_travellers. Also, assign the value of Number_of_buses as per the following conditions :

Number_of_travellers	**Number_of_buses**
Less than 20	1
Equal to or more than 20 and less than 40	2
Equal to 40 or more than 40	3

A function ShowPlan( ) to display the content of all the data members on screen.

(d) Answer the questions (i) to (iv) based on the following code: 4

```
class Medicines
{
        char Category[lO];
        char Date_of_manufacture[lO];
        char Company[20];
        public:
                Medicines();
                void entermedicinedetails();
                void showmedicinedetails();
} ;
class Capsules: public Medicines
{
        protected:
                char capsule_name[30];
                char Volume_label[20];
        public:
                float Price;
                Capsules();
                void entercapsuledetails();
                void showcapsuledetails();
};
class Antibiotics: public Capsule
{
        int Dosage_units;
        char Side_effects[20];
        int Use_within_days;
        public:
                Antibiotics() ;
                void enterdetails();
                void showdetails();
};
```

(i) How many bytes will be required by an object of class Medicines and an object of class Antibiotics respectively?

(ii) Write names of all the member functions accessible from the object of class Antibiotics.

(iii) Write names of all the members accessible from member functions of class Capsules.

(iv) Write names of all the data members, which are accessible from objects of class Antibiotics.

3. (a) Write a function in C++ which accepts an integer array and its size as arguments/parameters and exchanges the values of first half side elements with the second half side elements of the array. 3

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

(b) An array Arr[15][35] is stored in the memory along the column with each of its elements occupying 8 bytes. Find out the base address and the address of an element Arr[2][5], if the location Arr[5][10] is stored at the address 4000. 4

(c) Write a function in C++ to perform a PUSH operation in a dynamically allocated stack considering the following: 4

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

(d) Write a function in C++ to print the sum of all the values which are either divisible by 2 or are divisible by 3 present in a two-dimensional array passed as the argument to the function. 3

(e) Evaluate the following postfix notation of expression: 2

10 20 + 25 15 - * 30 /

4. (a) Observe the program segment given below carefully, and answer the question that follows: 1

```
class Book
{
        int Book no;
        char Book_name[20];
        public:
                //function to enter Book details
                void enterdetails();
                // function to display Book details
                void showdetails();
                //function to return Book_no
                int Rbook_no (){return Book_no;}
} ;
void Modify(Book NEW)
{
        fstream File;
        File.open("BOOK.DAT",ios::binary|ios::in|ios::out);
        Book OB;
        int Recordsread = 0, Found = 0;
        while (!Found && File.read((char*)&OB, sizeof (OB)))
        {
                Recordsread ++ ;
                if (NEW.RBook_no() = = OB.RBook_no))
                {
                        _____ //Missing Statement
                        File.write((char*)&NEW, sizeof (NEW));
                        Found = 1;
                }
```

```
            else
                File.write((char*)&OB, sizeof(OB));
        }
        if (! Found)
            cout<<" Record for modification does not exist";
        File.close();
}
```

If the function Modify( ) is supposed to modify a record in file BOOK.DAT with the values of Book NEW passed to its argument, write the appropriate statement for **Missing Statement** using seekp( ) or seekg( ), whichever needed, in the above code that would write the modified record at its proper place.

(b) Write a function in C++ to count and display the number of lines starting with alphabet 'A' present in a text file "LINES.TXT". 2

Example:
If the file "LINES.TXT" contains the following lines,
A boy is playing there.
There is a playground.
An aeroplane is in the sky.
Alphabets and numbers are allowed in the password.
The function should display the output as 3

(c) Given a binary file STUDENT.DAT, containing records of the following class Student type 3

```
class Student
{
    char S_Admno[lO]; //Admission number of student
    char S_Name[30]; //Name of student
    int Percentage; //Marks Percentage of student
    public:
        void EnterData()
        {
            gets(S_Admno);gets(S_Name);cin>>Percentage;
        }
        void DisplayData()
        {
            cout<<setw(12)<<S_Admno;
            cout<<setw(32)<<S_Name;
            cout<<setw(3)<<Percentage<<endl;
        }
        int ReturnPercentage(){return Percentage;}
};
```

Write a function in C++, that would read contents of file STUDENT.DAT and display the details of those Students whose Percentage is above 75.

5. (a) What do you understand by the terms **Primary Key** and **Degree of a relation** in relational database? 2

(b) Consider the following tables EMPLOYEES and EMPSALARY. Write SQL commands for the statements (i) to (iv) and give outputs for SQL queries (v) to (viii).

EMPLOYEES

EMPID	FIRSTNAME	LASTNAME	*ADDRESS*	CITY
010	George	Smith	83 First Street	Howard
105	Mary	Jones	842 Vine Ave.	Losantiville
152	Sam	Tones	33 Elm St.	Paris
215	Sarah	Ackerman	440 U.S. 110	Upton
244	Manila	Sengupta	24 Friends Street	New Delhi
300	Robert	Samuel	9 Fifth Cross	Washington
335	Henry	Williams	12 Moore Street	Boston
400	Rachel	Lee	121 Harrison St.	New York
441	Peter	Thompson	11 Red Road	Paris

EMPSALARY

EMPID	SALARY	BENEFITS	DESIGNATION
010	75000	15000	Manager
105	65000	15000	Manager
152	80000	25000	Director
215	75000	12500	Manager
244	50000	12000	Clerk
300	45000	10000	Clerk
335	40000	10000	Clerk
400	32000	7500	Salesman
441	28000	7500	Salesman

(i) To display Firstname, Lastname, Address and City of all employees living in Paris from the table EMPLOYEES.
(ii) To display the content of EMPLOYEES table in descending order of FIRSTNAME.
(iii) To display the Firstname, Lastname, and Total Salary of all Managers from the tables EMPLOYEES and EMPSALARY, where Total Salary is calculated as Salary + Benefits.
(iv) To display the Maximum salary among Managers and Clerks from the table EMPSALARY.

(v)
```
SELECT FIRSTNAME, SALARY
FROM EMPLOYEES, EMPSALARY
WHERE DESIGNATION = 'Salesman' AND
EMPLOYEES.EMPID=EMPSALARY.EMPID;
```

(vi)
```
SELECT COUNT (DISTINCT DESIGNATION)FROM EMPSALARY;
```

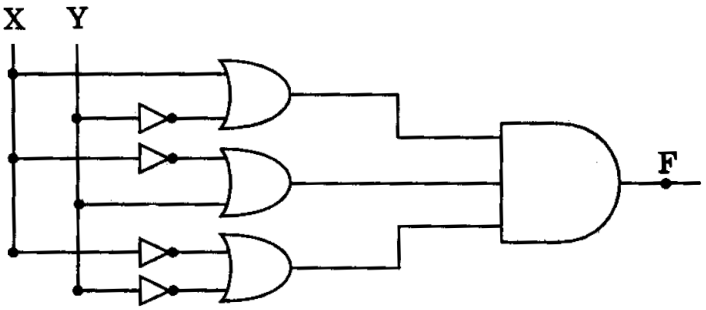
(vii)
```
SELECT DESIGNATION, SUM(SALARY)
FROM EMPSALARY
GROUP BY DESIGNATION HAVING COUNT (*)>2;
```

(viii)
```
SELECT SUM (BENEFITS)
FROM EMPLOYEES
```

WHERE DESIGNATION = 'Clerk';

6. (a) State and verify Associative law in Boolean Algebra. 2

(b) Write the equivalent Boolean expression for the following Logic Circuit : 2



(c) Write the SOP form of a Boolean Function F, which is represented by the following truth table: 1

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

(d) Reduce the following Boolean expression using K - Map: 3

F (A, B, C, D) = Π □(O, 1, 2, 3, 4, 5, 10, 11, 15)

7. (a) What is the difference between Message Switching technique and Packet Switching technique? 1

(b) Expand the following terminologies : 2

(i) TCP/IP

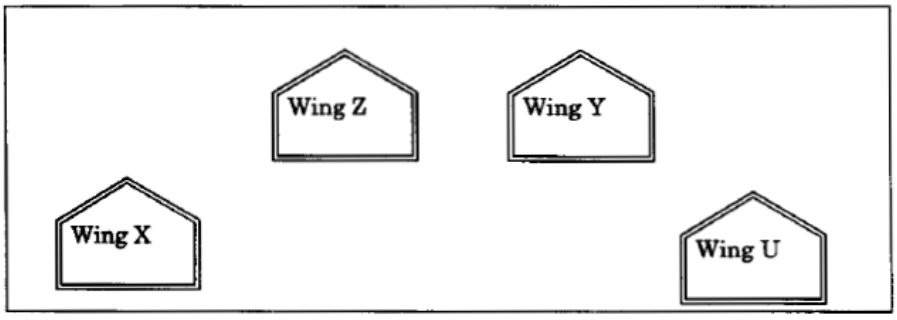
(ii) XML

(iii) CDMA

(iv) WLL

(c) Write two application of Cyber Law. 1

(d) The Great Brain Organisation has set up its new Branch at Srinagar for its office and web based activities. It has 4 Wings of buildings as shown in the diagram :



Center to center distances between various blocks

Wing X to Wing Z	50m
Wing Z to Wing Y	70m
Wing Y to Wing X	125m
Wing Y to Wing U	80m
Wing X to Wing U	175m
Wing Z to Wing U	90m

Number of Computers

Wing X	50
Wing Z	30
Wing Y	150
Wing U	15

(i) Suggest a most suitable cable layout of connections between the Wings, and topology. 1

(ii) Suggest the most suitable place (i.e. Wing) to house the server of this organisation with a suitable reason, with justification. 1

(iii) Suggest the placement of the following devices with justification: 1

(1) Repeater

(2) Hub/Switch

(iv) The organization is planning to link its head office situated in Delhi with the offices at Srinagar. Suggest an economic way to connect it; the company is ready to compromise on the speed of connectivity. Justify your answer. 1

Model Test Paper - 2

Time allowed : 3 hours *Maximum Marks : 70*

Instructions :

(i) ***All*** *questions are compulsory.*

(ii) Programming Language : C++

1. (a) Differentiate between a default constructor and copy constructor, giving suitable examples of each. 2

(b) Name the header files to which the following belong : 1

(i) puts ( )

(ii) isalnum ( )

(c) Rewrite the following program after removing the syntactical error(s), if any. Underline each correction. 2

```
#include <iostream.h>
const int Dividor 5;
void main()
```

```
{
        Number=15;
        for (int Count = l;Count=<5;Count++,Number-=3)
                if (Number%Dividor==0)
                        cout<<Number / Dividor;
                        cout<<endl;
                else
                        cout<<Number + Dividor<<endl;
}
```

(d) Find the output of the following program : 3

```
#include<iostream.h>
struct Package
{
        int Length, Breadth, Height;
};
void Occupies(Package M)
{
        cout<<M.Length<<"x"<<M.Breadth<<"x";
        cout<<M.Height<<endl;
}
void main()
{
        Package Pl={100,150,50}, P2, .P3;
        ++P1.Length;
        Occupies(P1);
        P3 = P1;
        ++P3.Breadth;
        P3.Breadth++;
        Occupies (P3);
        P2 = P3;
        P2.Breadth+=50;
        P2. Height--;
        Occupies(P2);
}
```

(e) Find the output of the following program : 2

```
#include<iostream.h>
#include<string.h>
#include<ctype.h>
void Change(char Msg[],int Len)
{
        for (int Count =0; Count<Len; Count++ )
        {
                if (islower(Msg[Count]))
                        Msg[Count]= toupper{Msg[Count]);
                else if (isupper(Msg[Count]))
                        Msg[Count]= tolower(Msg[Count]);
                else if (isdigit(Msg[Count]))
                        Msg[Count]= Msg[Count] + 1;
                else Msg[Count] = '*' ;
        }
}
void main()
{
        char Message [] = "2005 Tests ahead";
```

```
    int Size = strlen(Message);
    Change(Message,Size);
    cout<<Message<<endl;
    for (int C = 0,R=Size-l;C<=Size/2; C++,R--)
    {
        char Temp = Message[C];
        Message[C]= Message[R];
        Message[R]= Temp;
    }
    cout<<Message<<endl;
}
```

(f) Observe the following program GAME.CPP carefully, if the value of Num entered by the user is 14, choose the correct possible output(s) from the options from (i) to (iv), and justify your option. 2

```
//Program : GAME.CPP
#include<stdlib.h>
#include<iostream.h>
void main()
{
    randomize();
    int Num, Rndnum;
    cin>>Num;
    Rndnum = random (Num) + 7;
    for (int N = 1; N<=Rndnum ; N++)
        cout<<N<<" ";
}
```

Output Options :

(i) 1 2 3

(ii) 1 2 3 4 5 6 7 8 9 10 11

(iii) 1 2 3 4 5

(iv) 1 2 3 4

2. (a) Define the term Data Encapsulation in the context of Object Oriented Programming. Give a suitable example using a C++ code to illustrate the same. 2

(b) Answer the questions (i) and (ii) after going through the following class : 2

```
class Exam
{
    int Marks;
    char Subject[20];
    public:
        Exam () //Function 1
        {
            Marks = 0;
            strcpy (Subject,"Computer");
        }
        Exam(char S[]) //Function 2
        {
            Marks = 0;
            strcpy(Subject,S);
        }
        Exam(int M) //Function 3
        {
            Marks = M;
            strcpy(Subject,"Computer");
```

```
        }
        Exam(char S[], int M) //Function 4
        {
            Marks = M;
            strcpy (Subject,S);
        }
};
```

(i) Write statements in C++ that would execute Function 3 and Function 4 of class Exam.

(ii) Which feature of Object Oriented Programming is demonstrated using Function 1, Function 2, Function 3 and Function 4 in the above class Exam ?

(c) Define a class Travel in C++ with the following descriptions : 4

Private Members :

- TravelCode of type long
- Place of type character array (string)
- No_of_travellers of type integer
- No_of_buses of type integer

Public Members :

- A constructor to assign initial values of TravelCode as 201, Place as "Nainital", No_of_travellers as 10, No_of_buses as 1
- A function NewTravel() which allows user to enter TravelCode, Place and No_of_travellers. Also, assign the value of No_of_buses as per the following conditions :

No_of_travellers	**No_of_buses**
Less than 20	1
Equal to or more than 20 and less than 40	2
Equal to 40 or more than 40	3

- A function ShowTravel( ) to display the content from all the data members on screen.

(d) Answer the questions (i) to (iv) based on the following code : 4

```
class Drug
{
    char Category[10];
    char Date_of_manufacture[10];
    char Company[20];
    public:
        Drug();
        void enterdrugdetails();
        void showdrugdetails{};
};
class Tablet : public Drug
{
    protected:
        char tablet_name[30];
        char Volume_label[20];
    public:
        float Price;
        Tablet();
        void entertabletdetails();
        void showtabletdetails ();
};
class PainReliever : public Tablet
{
```

```
        int Dosage_units;
        char Side_effects[20];
        int Use_within_days;
        public:
                PainReliever();
                void enterdetails();
                void showdetails();
};
```

(i) How many bytes will be required by an object of class Drug and an object of class PainReliever respectively ?

(ii) Write names of all the data members which are accessible from the object of class PainReliever.

(iii) Write names of all the members accessible from member functions of class Tablet.

(iv) Write names of all the member functions which are accessible from objects of class PainReliever.

3. (a) Write a function in C++ which accepts an integer array and its size as arguments/parameters and exchanges the values of first half side elements with the second half side elements of the array. 3

Example :

If an array of eight elements has initial content as

8, 10, 1, 3, 17, 90, 13, 60

The function should rearrange the array as

17, 90, 13, 60, 8, 10, 1, 3

(b) An array Arr[35][15] is stored in the memory along the row with each of its element occupying 4 bytes. Find out the base address and the address of an element Arr[20][5], if the location Arr[2][2] is stored at the address 3000. 4

(c) Write a function in C++ to perform a DELETE operation in a dynamically allocated queue considering the following description : 4

```
struct Node
{
        float U,V;
        Node *Link;
};
class QUEUE
{
        Node *Rear,*Front;
        public:
                QUEUE(){Rear=NULL;Front=NULL;}
                void INSERT();
                void DELETE();
                ~QUEUE();
};
```

(d) Write a function in C++ to print the sum of all the values which are either divisible by 3 or are divisible by 5 present in a two dimensional array passed as the argument to the function. 3

(e) Evaluate the following postfix notation of expression :

20 10 + 5 2 * – 10 / 2

4. (a) Observe the program segment given below carefully, and answer the question that follows:

```
class Member
{
        int Member_no;
        char Member_name[20];
        public :
                //function to enter Member details
                void enterdetails{) ;
                // function to display Member details
                void showdetails();
                //function to return Member_no
                int RMember_no() {return Member_no; }
};
void Update(Member NEW)
{
        fstream File;
        File.open("MEMBER.DAT",ios::binary|ios::in|ios::out);
        Member OM;
        int Recordsread = 0, Found = 0;
        while (!Found && File.read((char*)&OM, sizeof(OM)))
        {
                Recordsread ++;
                if (NEW.RMember_no() == OM.RMember_no())
                {
                        _____//Missing Statement
                        File.write((char*)&NEW, sizeof(NEW));
                        Found = 1;
                }
                else
                        File.write((char*)&OM, sizeof(OM));
        }
        if (!Found)
                cout<<"Record for modification does not exist";
        File.close();
}
```

If the function Update ( ) is supposed to modify a record in file MEMBER.DAT with the values of Member NEW passed to its argument, write the appropriate statement for **Missing Statement** using seekp( ) or seekg( ), whichever needed, in the above code that would write the modified record at its proper place.

(b) Write a function in C++ to count and display the number of lines not starting with alphabet 'A' present in a text file 'STORY.TXT".

2

Example :

If the file "STORY.TXT" contains the following lines,

The rose is red.
A girl is playing there.
There is a playground.
An aeroplane is in the sky.
Numbers are not allowed in the password.

The function should display the output as 3

(c) Given a binary file APPLY.DAT, containing records of the following class Applicant type 3

```
class Applicant
{
        char A_Rno[10]; //Roll number of applicant
        char A_Name[30]; //Name of applicant
        int A_Score; //Score of applicant
        public:
                void Enrol()
                {
                        gets (A_Rno); gets (A_Name) ; cin>>A_Score;
                }
                void Status()
                {
                        cout<<setw(12)<<A_Admno;
                        cout<<setw(32)<<A_Name;
                        cout<<setw(3)<<A_Score<<endl;
                }
                int ReturnScore(){return A_Score;}
};
```

Write a function in C++, that would read contents of file APPLY.DAT and display the details of those Students whose A_ Score is below 70.

5. (a) What do you understand by the terms **Candidate Key** and **Cardinality of a relation** in relational database ? 2

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

WORKERS

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

DESIG

W_ID	SALARY	BENEFITS	DESIGNATION
102	75000	15000	Manager
105	85000	25000	Director
144	70000	15000	Manager
210	75000	12500	Manager
255	50000	12000	Clerk
300	45000	10000	Clerk
335	40000	10000	Clerk
400	32000	7500	Salesman
451	28000	7500	Salesman

(i) To display W_ID, Firstname, Address and City of all employees living in New York from the table WORKERS.

(ii) To display the content of WORKERS table in ascending order of LASTNAME.

(iii) To display the Firstname, Lastname, and Total Salary of all Clerks from the tables WORKERS and DESIG, where Total Salary is calculated as Salary + Benefits.

(iv) To display the Minimum salary among Managers and Clerks from the table DESIG.

(v)
```
SELECT FIRSTNAME, SALARY
FROM WORKERS, DESIG
WHERE DESIGNATION = 'Manager' AND
WORKERS.W_ID=DESIG.W_ID;
```

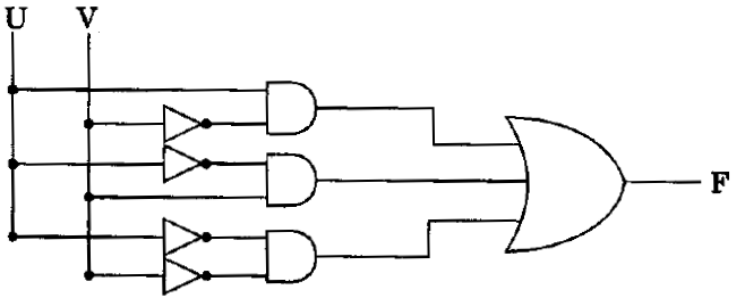
(vi)
```
SELECT COUNT(DISTINCT DESIGNATION) FROM DESIG;
```

(vii)
```
SELECT DESIGNATION, SUM(SALARY)
FROM DESIG
GROUP BY DESIGNATION HAVING COUNT(*)<3;
```

(viii)
```
SELECT SUM(BENEFITS)
FROM WORKERS
WHERE DESIGNATION = 'Salesman';
```

6. (a) State and verify Absorption law in Boolean Algebra. 2

(b) Write the equivalent Boolean expression for the following Logic Circuit : 2



(c) Write the POS form of a Boolean Function F, which is represented by the following truth table : 1

X	Y	Z	F
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	00
1	1	1	

(d) Reduce the following Boolean expression using K - Map : 3

F (A, B, C, D) = ∑□(0, 1, 2, 3, 4, 5, 10, 11, 15)

7. (a) Compare Optical Fiber and Coaxial transmission media. 1

(b) Expand the following terminologies : 1

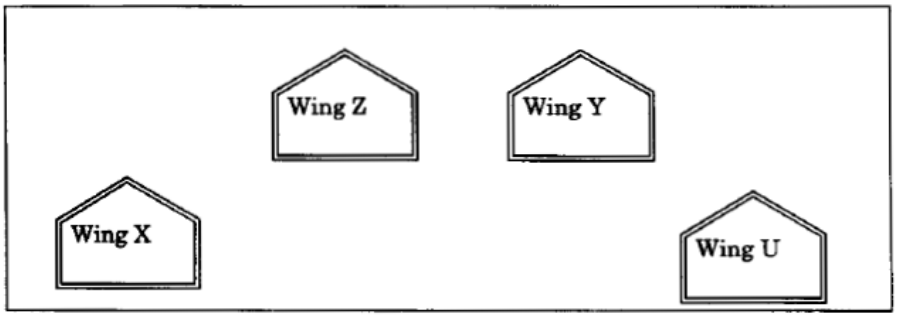
(i) HTML

(ii) GSM

(c) What is the difference between XML and HTML ? Write two differences. 1

(d) What do you understand by the terms Cookies and Firewall ? 1

(e) The Cyber Mind Organisation has set up its new Branch at Mizoram for its office and web based activities. It has 4 Wings of buildings as shown in the diagram :



Center to center distance between various blocks

Wing X to Wing Z	40m
Wing Z to Wing Y	60m
Wing Y to Wing X	135m
Wing Y to Wing U	70m
Wing X to Wing U	165m
Wing Z to Wing U	80m

Number of Computers

Wing X	50
Wing Z	130
Wing Y	140
Wing U	15

(el) Suggest a most suitable cable layout of connections between the Wings and topology. 1
(e2) Suggest the most suitable place (i.e. Wing) to house the server of this organization with a suitable reason with justification. 1
(e3) Suggest the placement of the following devices with justification : 1
(i) Repeater
(ii) Hub/Switch
(e4) The organization is planning to link its head office situated in Delhi with the offices as Mizoram. Suggest an economic way to connect it; the company is ready to compromise on the speed of connectivity. Justify your answer.
1

Model Test Paper - 3

Time allowed : 3 hours *Maximum Marks : 70*

Instructions :

(i) ***All*** *questions are compulsory.*

(ii) Programming Language : C++

1. (a) Differentiate between a Run Time Error and Syntax Error. Also give suitable examples of each in C++. 2

(b) Name the header file(s) that shall be needed for successful compilation of the following C++ code. 1

```
void main ( )
{
        char String [20];
        gets (String);
        strcat (String, "CBSE");
        puts (String);
}
```

(c) Rewrite the following program after removing the syntactical error(s) if any. Underline each correction. 2

```
# include <iostream.h>
const int Max 10;
void main ( )
{
        int Numbers [Max];
        Numbers = { 20, 50,10, 30,40 } ;
        for (Loc= Max-1 ; Loc > = 0 ; Loc - -)
                cout>>Numbers [Loc];
}
```

(d) Find the output of the following program : 2

```
# include < iostream.h>
void main ()
{
        intArray[] = {4,6,10,12};
        int *pointer = Array ;
        for (int I=1 ; I<=3 ; I++)
        {
                cout<<*pointer<<#";
                pointer ++;
        }
        cout<<endl;
        for (I=1 ; I<=4 ; I++)
        {
                (*pointer)*=3 ;
                -- pointer;
        }
        for(I=l; I<5; I + + )
                cout << Array [I-1] << "@";
        cout << endl;
}
```

(e) Find the output of the following program : 3

```
# include < iostream.h>
void Withdef (int HisNum = 30)
{
        for (int 1=20 ; I<*= HisNum; I+=5)
                cout<<I<<"";
        cout<<endl;
}
void Control (int &MyNum)
{
        MyNum+=10;
        Withdef(MyNum);
}
void main ()
{
        int YourNum=20;
        Control (YourNum);
        Withdef();
        cout<<"Number="<<YourNum<<endl;
}
```

(f) In the following C++ program what is the expected value of MyMarks from Options (i) to (iv) given below. Justify answer. 2

```
#include<stdlib. h >
# include<iostream. h>
void main ()
{
        randomize ();
        int Marks [ ]= {99, 92, 94, 96, 93, 95}, MyMarks;
        MyMarks = Marks [1 + random (2) ];
        cout<<MyMarks<<endl;
}
```

(i) 99 (ii) 94
(iii) 96 (iv) None of the above

2. (a) Differentiate between Constructor and Destructor function in context of Classes and Objects using **C++** 2

(b) Answer the questions (i) and (ii) after going through the following class 2

```
class Maths
{
        char Chapter [20];
        int Marks;
        public:
        Maths ( ) //Member Function 1
        {
                strcpy (Chapter, "Geometry");
                Marks = 10;
                cout<<"Chapter Initialised";
        }
        ~Math () //Member Function 2
        {
```

```
        cout<<"Chapter Over";
    }
};
```

(i) Name the specific features of class shown by Member Function 1 and Member Function 2 in the above example.
(ii) How would Member Function 1 and Member Function 2 get executed?
(c) Define a class Tour in C++ with the description given below : 3
Private Members :

TCode	of type string
NoofAdults	of type integer
NoofKids	of type integer
Kilometres	of type integer
TotalFare	of type float

Public Members :
• A constructor to assign initial values as follows :
TCode with the word "NULL"
NoofAdults as 0
NoofKids as 0
Kilometres as 0
TotalFare as 0
• A function AssignFare ( ) which calculates and assigns the value of the data member TotalFare as follows
For **each** Adult

Fare(Rs)	**For Kilometres**
500	>=1000
300	<1000&>=500
200	<500

For **each** Kid the above Fare will be 50% of the Fare mentioned in the above table
For example :
If Kilometres is 850, NoofAdults = 2 and NoofKids = 3
Then TotalFare should be calculated as
NumofAdults * 300 + NoofKids * 150
i.e. 2*300 + 3*150=1050
• A function EnterTour( ) to input the values of the data members TCode, NoofAdults, NoofKids and Kilometres; and invoke the Assign Fare( ) function.
• A function ShowTour( ) which displays the content of all the data members for a Tour.
(d) Answer the questions (i) to (iv) based on the following code : 4

```
class Trainer
{
    char TNo [5], TName [20], Specialisation [10];
    int Days;
    protected :
        float Remuneration;
        void AssignRem (float);
    public :
        Trainer ( ) ;
        void TEntry ( );
        void TDisplay ( );
};
```

```
class Learner
{
        char Regno [10], LName [20], Program [10];
        Protected :
                int Attendance, Grade;
        public:
                Learner ( );
                void LEntry ( );
                void LDisplay ( );
};
class Institute : public Learner, public Trainer
{
        char ICode[10], IName [20];
        public:
                Institute ( );
                void IEntry ( );
                void IDisplay ( );
};
```

(i) Which type of Inheritance is depicted by the above example?

(ii) Identify the member function(s) that cannot be called directly from the objects of class Institute from the following

TEntry( )
LDisplay()
IEntry()

(iii) Write name of all the member(s) accessible from member functions of class Institute.

(iv) If class Institute was derived privately from class Learner and privately from class Trainer, then, name the member function(s) that could be accessed through Objects of class Institute.

3. (a) Write a function in C++ which accepts an integer array and its size as arguments and replaces elements having odd values with thrice its value and elements having even values with twice its value.

Example : if an array of five elements initially contains the elements as

3, 4, 5, 16, 9

then the function should rearrange the content of the array as

9, 8, 15, 32, 27 4

(b) An array Array[20][15] is stored in the memory along the column with each element occupying 8 bytes. Find out the Base Address and address of the element Array[2][3] if the element Array [4] [5] is stored at the address 1000. 4

(c) Write a function in C++ to delete a node containing Book's information, from a dynamically allocated Stack of Books implemented with the help of the following structure. 4

```
struct Book
{
        int BNo;
        char BName[20];
        Book *Next;
};
```

(d) Write a function in C++ which accepts a 2D array of integers and its size as arguments and displays the elements which lie on diagonals. 2

[Assuming the 2D Array to be a square matrix with odd dimension
i.e. 3×3, 5×5, 7×7 etc.]
Example, if the array content is

5 4 3
6 7 8
1 2 9

Output through the function should be :
Diagonal One : 5 7 9
Diagonal Two : 3 7 1

(e) Evaluate the following postfix notation of expression : 2

25 8 3 - / 6 * 10 +

4. (a) Observe the program segment given below carefully, and answer the question that follows: 1

```
class PracFile
{
        intPracno;
        char PracName[20];
        int TimeTaken;
        int Marks;
        public:
                // function to enter PracFile details
                void EnterPrac( );
                // function to display PracFile details
                void ShowPrac( ):
                // function to return TimeTaken
                int RTime() {return TimeTaken;}
                // function to assign Marks
                void Assignmarks (int M)
                { Marks = M;}
};
void AllocateMarks( )
{
        fstreamFile;
        File.open("MARKS.DAT",ios::binary|ios::in|ios::out);
        PracFile P;
        int Record = 0;
        while (File.read(( char*) &P, sizeof(P)))
        {
                if(P.RTime()>50)
                        P.Assignmarks(0)
                else
                        P.Assignmarks(10)
                _____ //statement 1
                _____ //statement 2
                Record + + ;
        }
        File.close();
}
```

If the function AllocateMarks () is supposed to Allocate Marks for the records in the file MARKS.DAT based on their value of the member TimeTaken. Write C++ statements for the

statement 1 and **statement 2,** where, **statement 1** is required to position the file write pointer to an appropriate place in the file and **statement 2** is to perform the write operation with the modified record.

(b) Write afunction in C++ to print the count of the word is as an independent word in at text file DIALOGUE.TXT.

2

For example, if the content of the file DIALOGUE. TXT is

This is his book. Is this book good?

Then the output of the program should be 2.

(c) Given a binary file GAME.DAT, containing records of the following structure type 3

```
struct Game
{
        char GameName [20];
        char Participant [10] [30];
};
```

Write a function in C++ that would read contents from the file GAME.DAT and creates a file named BASKET.DAT copying only those records from GAME.DAT where the game name is "Basket Ball".

5. (a) Differentiate between primary key and alternate key. 2

(b) Consider the following tables. Write SQL commands for the statements

(i) to (iv) and give outputs for SQL queries (v) to (viii) 6

TABLE:SENDER

SenderlD	SenderName	***SenderAddress***	**SenderCiry**
ND01	R Jain	2, ABC Appts	New Delhi
MU02	H Sinha	12, Newtown	Mumbai
MU15	S Jha	27/A, Park Street	Mumbai
ND50	T Prasad	122-K, SDA	New Delhi

TABLE : RECIPIENT

RecID	**SenderlD**	**RecName**	**RecAddress**	**RecCity**
KO05	ND01	R Bajpayee	5, Central Avenue	Kolkata
ND08	MU02	S Mahajan	116, A Vihar	New Delhi
MU19	ND01	H Singh	2A, Andheri East	Mumbai
MU32	MU15	P K Swamy	B5, C S Terminus	Mumbai
ND48	ND50	S Tripathi	13, B1 D, Mayur Vihar	New Delhi

(i) To display the names of all Senders from Mumbai

(ii) To display the RecID), SenderName, SenderAddress, RecName, RecAddress for every Recipient

(iii) To display Recipient details in ascending order of RecName

(iv) To display number of Recipients from each city

(v) SELECT DISTINCT SenderCity FROM Sender;

(vi)
```
SELECT A. SenderName, B.RecName
FROM Sender A, Recipient B
WHERE A. SenderlD = B.SenderlD AND B.RecCity = 'Mumbai';
```

(vii)
```
SELECT RecName, RecAddress
FROM Recipient
WHERE RecCity NOT IN ('Mumbai', 'Kolkata');
```

(viii) SELECT RecID, RecName
FROM Recipient
WHERE SenderID='MU02' ORSenderID='ND50';

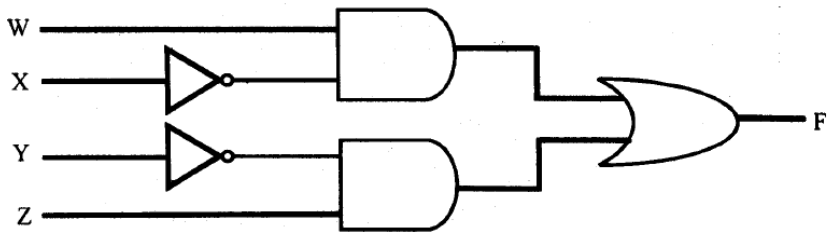
6. (a) State Distributive law and verify the same using truth table. 2

(b) Write the equivalent Canonical Sum of Product expression for the following Product of Sum Expression

2

F(X,Y,Z) = □Π□(1,3,6,7)

(c) Write the equivalent Boolean Expression for the following Logic Circuit. 2



(d) Reduce the following Boolean expression using K-Map 2

F(U,V,W,Z) = Σ□(0, 1, 2, 3, 4, 10, 11)

7. (a) What is the significance of Cyber law ? 1

(b) Expand the following terms with respect to Networking : 2

(i) CDMA (iii) FTP

(ii) WLL (iv) HTML

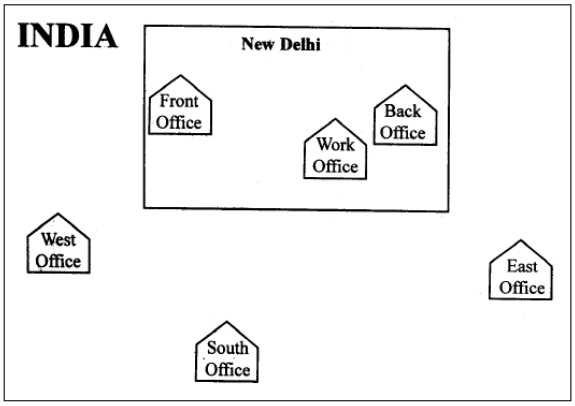
(c) Which of the following unit measures the speed with which data can be transmitted from one node to another node of a network? Also, give the expansion of the suggested unit. 1

(i) Mbps

(ii) KMph

(iii) MGps

(d) "Bhartiya Connectivity Association" is planning to spread their offices in four major cities in India to provide regional IT infrastructure support in the field of Education & Culture. The company has planned to setup their head office in New Delhi in three locations and have named their New Delhi offices as "Front Office", "Back Office" and "Work Office". The company has three more regional offices as "South Office", "East Office" and "West Office" located in other three major cities of India. A rough layout of the same is as follows : 4



Approximate distances between these offices as per network survey team is as follows:

Place From	Place To	Distance
BackOffice	Front Office	10KM
Back Office	Work Office	70 Meter
Back Office	East Office	1291 KM
BackOffice	West Office	790 KM
Back Office	South Office	1952 KM

In continuation of the above, the company experts have planned to install the following number of computers in each of their offices :

Back Office	100
Front Office	20
Work Office	50
East Office	50
West Office	50
South Office	50

(i) Suggest network type (out of LAN, MAN, WAN) for connecting each of the following set of their offices :

• Back Office and Work Office

• Back Office and South Office

(ii) Which device you will suggest to be procured by the company for connecting all the computers with in each of their offices out of the following devices?

• Switch/Hub

• Modem

• Telephone

(iii) Which of the following communication medium, you will suggest to be procured by the company for connecting their local offices in New Delhi for very effective and fast communication?

• Telephone Cable

• Optical Fiber

• Ethernet Cable

(iv) Suggest a cable/wiring layout for connecting the company's local offices located in New Delhi. Also, suggest an effective method/technology for connecting the company's regional offices-"East Office", "West Office" and "South Office" with offices located in New Delhi.

Model Test Paper - 4

Time allowed : 3 hours *Maximum Marks : 70*

Instructions :

(i) ***All*** *questions are compulsory.*

(ii) Programming Language : C++

1. (a) Define any two features of OOPs. Also give suitable example in C++. 2

2. (b) Name the Header file(s) that shall be needed for successful compilation of the following C++ code

```
void main()
{
        int a[10];
        for(int i=0;i<10;i++)
        {
        cin>>a[i];
        if(a[i]%2==0)
                a[i]=pow(a[i],3);
        else
                a[i]=sqrt(a[i]);
        if(a[i]>32767)
                exit(0);
        }
        getch();
}
```

1

(c) Rewrite the following program after removing syntactical error(s) if any. Underline each correction. 2

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

(d) Find the output of the following program : 2

```
#include<iostream.h>
#include<string.h>
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;
            }
        }
```

(e) Find the output of the following program 2

```
#include<iostream.h>
class student
{
        public:
        student()
        {
                cout<<"\n Computer Science";
        }
        ~student()
        {
                cout<<"  subject";
        }
}st;
void main()
{
        cout<<" is my best"
}
```

(f) In the following C++ program , what will the maximum and minimum value of r generated with the help of random function. 2

```
#include<iostream.h>
#include<stdlib.h>
void main()
{
        int r;

        randomize();
        r=random(20)+random(2);
        cout<<r;
}
```

(g) Define macro with a suitable example. 2

2. (a) Differentiate between a Constructor and Destructor in context of class and object . Give suitable example in C++. 2

(b) Answer the questions (i) and (ii) after going through the following class : 2

```
class Computer
{
        char C_name[20];
        char Config[100];
        public:
                Computer(Computer &obj); // function1
                Computer();              //function 2
                Computer(char *n,char *C); // function3
};
```

(i) Complete the definition of the function 1.

(ii) Name the specific feature of the OOPs shown in the above example.

(c) Define a class Student in C++ with the description given below : 4

private members

rno	integer
name	array of 40 characters
address	array of 40 characters
marks	array of 5 integers
percentage	float variable
calper()	a function which will calculate & return the percentage of a student.

public members

init()	function to ask and store the values of rno, name, address and marks in 5 subjects.
display()	function to which will invoke calper () and display the details of the item in the following format :

MARK SHEET

Roll No : <Roll no of student>
Name : <Student Name >
Address : <ADDRESS >
Marks : <subject1 marks><subject 2 marks ><….subject 5 marks>
Percentage : <percentage of the student>

Also create main() function which will invoke all the public member functions.

(d) Answer the questions (i) to (iv) based on the following code : 4

```
class Employee
{
        int id;
        protected :
        char name[20];
        char doj[20];
        public :
        Employee();
        ~Employee();
        void get();
        void show();
};

class Daily_wager : protected Employee
{
        int wphour;
        protected :
        int nofhworked;
        public :
        void getd();
        void showd();
};
class Payment : private Daily_wager
{
        char date[10];
        protected :
        int amount;
```

```
        public :
        Payment();
        ~Payment();
        void show();
};
```

(i) Name the type of Inheritance depicted in the above example.

(ii) Name the member functions accessible through the object of class Payment.

(iii) From the following, Identify the member function(s) that can be called directly from the object of class Daily_wager class

```
show()
getd()
get()
```

(iv) Name the base & derived class of Daily_wager class.

3. (a) Write a function in C++ which accepts a integer array and its size as an arguments and prints the output (using nested loops) in following format :
Example : if the array is having
1 2 4 5 9
Then the output should be
1
2 2
4 4 4 4
5 5 5 5 5
9 9 9 9 9 9 4

(b) An array A[10][20] is stored in the memory with each element occupying 2 bytes of storage. If the Base address of array in the memory is 800 , determine the location of A[9][10] when the array is stored as (i) Row Major (ii) column major.

(c) Write a function in C++ to delete a node containing names of student , from a dynamically allocated Queue of names implemented with the help of following structure : 3

```
struct student
{
        char name[20];
        student *next;

}*front , *rear;
```

(d) Consider the following portion of a program , which implements a linked stack for Library . Write the definition of function PUSH(),to insert a new node in the stack with required information 3

```
struct Library
{
        int id;
        char names[20];
};
class stack
{
        Library *top;
        public :
        stack()
        {
        top=NULL;
        }
```

```
        void PUSH();
        void POP();
};
```

(e) Convert the following infix expression into postfix. show the stack status after execution of each operation: 2

TRUE OR FALSE AND NOT FALSE OR FALSE

4. (a) Differentiate between ios::app and ios::ate file opening modes.

(b) Write a function in C++ which will print the size of a text file "story.txt" in the form of bytes. 2

(c) Write a function in C++ which will increase the qty of a particular type of item from the file "stock.dat" . Assuming that the binary file is containing the records of following structure : 3

```
struct Products
{
        int id;
        char Iname[30];
        int type;
        int qty;
};
```

Accept the item type from user whose qty has to be increased .

5. (a) What do you understand by Primary Key. 2

(b) Consider the following tables Employee and salary. Write SQL commands for the statements (i) to (iv) and give outputs for SQL queries (v) to (viii) 6

Table : Employee

Eid	Name	Deptid	Qualification	Sex
1	Deepali Gupta	101	MCA	F
2	Rajat Tyagi	101	BCA	M
3	Hari Mohan	102	B.A	M
4	Harry	102	M.A	M
5	Sumit Mittal	103	B.Tech	M
6	Jyoti	101	M.Tech	F

Table : Salary

Eid	Basic	DA	HRA	Bonus
1	6000	2000	2300	200
2	2000	300	300	30
3	1000	300	300	40
4	1500	390	490	30
5	8000	900	900	80
6	10000	300	490	89

(i) To display the frequency of employees department wise.

(ii) To list the names of those employees only whose name starts with 'H'

(iii) To add a new column in salary table . the column name is total_sal.

(iv) To store the corresponding values in the total_sal column.

(v) Select name from employee where eid=(select eid from salary where basic= (select max(basic) from salary));

(vi) select max(basic) from salary where bonus >40;
(vii) Select count(*) from employee group by sex;
(viii) select Distinct deptid from Employee;

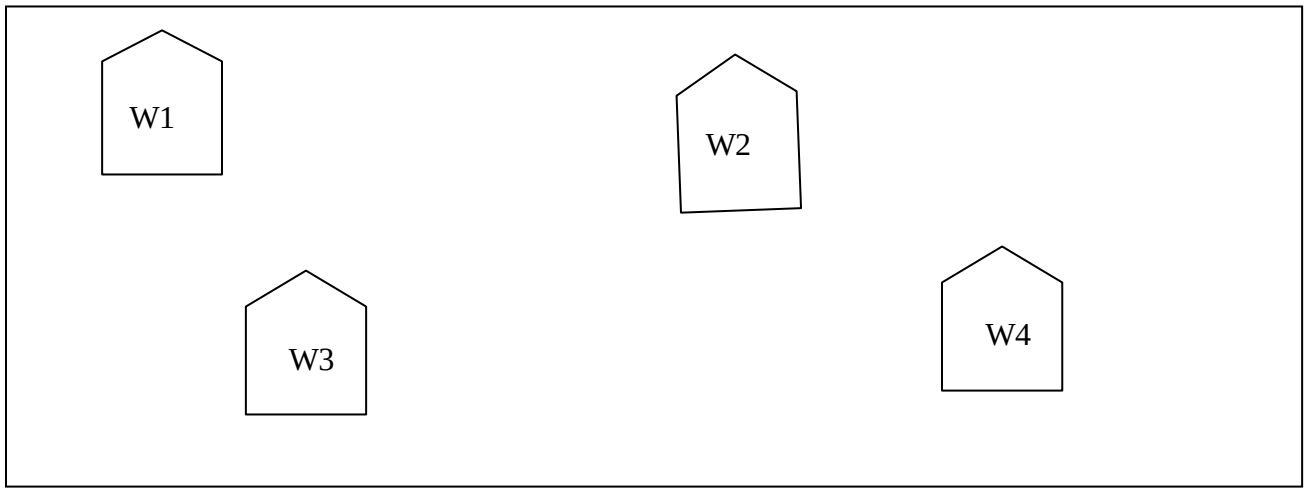
6. (a) State and prove the Distributive law algebraically. 2

7. (b) Write the equivalent POS expression of following SOP form
F (x,y,z)= Σ (0,2,4,6) 2

(c) Draw the Logical circuit of the following expression with the help of NAND gate only 1
x+yz

(d) Obtain the simplified form of a Boolean expression using K-Map.
F(a,b,c,d)=Σ(0,1,2,3,4,7,11,12,14) 3

7. (a) What do your understand by Hackers? 1
(b) Differentiate between Internet & Intranet 1
(c) Expand the following terminology :
(i) SMS (ii) FTP 1
(d) Define Repeater. 1
(e) A company in Reliance has 4 wings of buildings as shown in the diagram:



Center to center distances between various Buildings:

W3 to W1	50m
W1 to W2	60m
W2 to W4	25m
W4 to W3	170m
W3 to W2	125m
W1 to w4	90m

Number of computers in each of the wing:

W1	150
W2	15
W3	15

W4 25

Computers in each wing are networked but wings are not networked. The company has now decided to connect the wings also.

i) Suggest a most suitable cable layout & topology of the connection between the wings. [1]

ii) The company wants internet accessibility in all the wings. Suggest an economic technology . [1]

iii) Suggest the placement of the following devices with justification if the company wants minimized network traffic : [1]

1)Repeater
2) Hub
3) Switch
4) Bridge

iv) The company is planning to link its head office situated in India with the offices at Reliance. Suggest a way to connect it; the company does not want to compromise with the speed of connectivity. Justify your answer. [1]

Model Test Paper-5
Computer Science(083)
(Solved Paper)

1. (a) Differentiate between a run time error and syntax error. Also give suitable examples of each in C+ +.

Ans : Run time error: error occurring in a program during its execution. Program execution halts when such a error is encountered .

Example :

```
int   A,B,C;
cin >>A>>B;
C=A/B;//Run time error if value of b is zero.
```

Syntax error :
Error occurred due to wrong syntax of language deducted by the complier during compilation.

Example :

```
cout>>"a  c+ +  program";
```

(b) Name the header file(s) that shall be needed for successful compilation of the following C++ code

```
void main ( )
{
        char string [20];
        gets (string);
        strcat(String, CBSE);
        puts (string);
}
```

Ans : stdio.h

string.h

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

```
# include <iostream.h>
const int max 10;
void main ( )
{       int Number [max];
Numbers = {20, 50, 10, 30, 40};
for (Loc = Max-1; Loc> =0; Loc--)
cout >> Numbers [Loc];                }
```

Ans :

```
#include<iostream.h>
const int max +10;     //OR     const int MAX =5;
 void main ( )
```

```
{
 int numbers[max]={20,50,10,30,40};
int Loc;
for (int Loc= max-1;Loc>=0; loc--)
 cout<<Numbers[Loc];
}
```

(d) Find the output of the following program :

```
#include<iostream.h>
void main ( )
{
        int array [ ]={4,6,10,12}
        int *pointer =  array;
        For(int I =I ; I <3  ;  I++)
{       count<< *pointer <<``#``;
}
count<<end1;
for(I=1  I  <  4 ; I++)
{
          (*pointer) *=3 ;
}
for ( I=1;  I<5 ;  I++)
    cout<<array[I-1]<<``@``;cout<<end1;
```

Ans :

```
4#6#10#
12@18@30@36@
```

(e) Find the output of the following program:

```
#include<iostream.h>
{
void withdef(int hisnum; I+=5)
{          for  (int I=20; I<=hiusnum;  I+=5)
             cout<<I<<``,`` ;
             cout<<endl;                    }
          void control(int &mynum) ;   }
          void main( )
{          int yourNum=20;
           control(yourNum) ;
           withdef( ) ;
           cout<<``number=``<<yourNum<<endl;              }
```

Ans :

```
20,25,30,
20,25,30,
Number=30
```

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

```
#include<stdlib.h>
#include<iostream.h>
void main( )
{ randomize( );
        int Marks[ ] = { 99, 92, 94, 96, 93, 95}, Mymarks;
        MyMarks= marks[1+random(2)];
        cout<<MyMarks<<endl;
}
```

(i) 99 (ii) 94
(iii) 96 (iv) None of the above

Ans. (ii) 94

2. (a) Differentiate between constructor and destructor function in context of classes and objects using C++

Ans :

Constructor :

1. Name of the constructor functions is same as the name of the class.
2. No return type required for constructor functions.
3. Constructor functions are called automatically at the time of creation of the object.
4. Constructor can be overloaded.
5. constructor functions are defined in public.

Destructor :

1. Name of the destructor is same as the name of the class preceded by.
2. No return type required for destructor function.
3. Destructor functions are called automatically when the scope of the object gets over.
4. Destructor can not be overloaded.
5. Destructor function is defined in public.

(b) Answer the questions (i) and (ii) after going through the following class :

```
Class maths
{
        char chapter [20];
        int Marks [20]
       public:
Maths ( )
{
strcpy (chapter, ``geometery''
```

```
Marks =10;
cout <<``chapter Intialised ";
}
~Maths ( )                    //member Function 2
{
cout<<`` chapter over "
}
};
```

(i) Name the specific features of class shown by member function1 and member function 2 in above example?

Ans : FUNTION 1: constructor OR default constructor

(ii) How would Member Function 1 and Member Function 2 get get executed?

Ans. Function 1 is executed or invoked automatically when an object of class Maths is created.
Function 2 is invoked automatically when the scope of an object of class Maths comes to an end.

OR

Example :

```
{
Maths S1;
…..
}                          //constructor is invoked
}                         //Destructor is invoked
```

(c) Define a class tour in C++ with the description given below :

Private Members:

TCode	of type string
Noofadults	of type integer
Noofkids	of type integer
Kilometers	of type integer
Totalfare	of type float

Public Members :

- A constructor to assign initial values as follows:
  TCode with the word ``null``

Noofadults as0
Noof kids as0
Kilometers as 0

TotalFare as 0

- A function assign fare ( ) which calculates and assign

The value of the data member Total Fare as follows

For each adult

Fare (Rs)	For kilometers
500	>=1000
300	<1000 & >=500
200	<500

For each Kid the above Fare will be 50% of the fare mentioned in the above table

For example:

If Kilometers is 850, Noofadults= 2 and Noofkids=3

Then TotalFare shouls be calculated as

Num of adults* 300 +Noofkids*150

i.e. 2* 300+ 3* 150 = 1050

- A function entered to ( ) to input the values of the data members TCode, NoofAdults, NoofKids and Kilometers;and invoke the AssignFare( ) function.
- A function ShowTour( ) which displays the content of all the data members for a Tour.

Ans.

```
Class tour
{
    Char TCode [10];  or char *Tcode;
    int NoofAdults;
    int NoofKids;
    int KILometers;
    float TotalFare;
    public:
    tour ( )
{
    Strcpy (TCode,``NULL"):    OR TCode [0]=`\0'  OR strcpy(TCode, ``\0")
    Noofadults = 0;
  Noof kids  = 0;
  Kilometers = 0;
  TotalFare  = 0                };

Void tour: : assignfare ( )
{
if (kilometers>=1000)
    Totalfare =500* Noofadults+250* Noof kids  ;
     else if (kilometers>=500)
    Totalfare =300* Noofadults+150*Noofkids;
    else
  Total fare = 200* Noofadults+100*Noofkids;
```

```
}
void tour : : enter ( )
{
  gets ( TCode ); //or cin >> TCode;
  cin>>Noofadults>>Noofkids>>kilometers;
  Assignfare( );
}
  void tour::show tour( )
  {cout<<tcode<<Noofkids<<kilometers <<total fare<<end1;
}
```

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

```
Class trainer
{char  TNo [20],specializations [10];
 int days;
Protected:
  float  remuneration ;void assignrem(float);
Public:
  Trainer( );
        void TEntry( );
        void TDisplay( );
};
        class Learner
{
        char Regno[10],LName[20],Program[10];
protected:
        int Attendance, grade;
public:
        learner( );
        void TEntry( );
        void TDisplay( );
};
class institute : public Learner, Public Trainer
{
        char ICode[10],IName[20];
 public:
       Intitute ( ) ;
       void Ientry ( ) ;
       voidIdisplay( );
};
```

(i) Which type of inheritance is depicted by the above example?

Ans : Multiple Inheritance

(ii) Identify the member function (s) that cannot be called directly from the objects of the class institute from the following

Tentry( )
LDisplay( )
IEntry( )

Ans : **None** **Or** All the above functions can be called.

(iii) Write the names of all the member(s) accessible from member functions of class Institute.

Ans. DATA MEMBERS : ICode, IName, Attendance, Grade, Renumeration
MEMBER FUNCTIONS : IEntry( ), IDisplay( ), LEntry( ), LDisplay( ), AssignRem( ), TEntry( ), TDisplay( )

(iv) If class Institute was derived privately from class Learner and privately from class Trainer, then name the member function(s) that could be accessed through Objects of Class Institute

Ans. IEntry( ) , IDisplay( )

3. (a) Write a function in C++ which accepts an integer array and its size as arguments and replaces elements having odd values with thrice its value and elements having even values with twice its value.

Example : If an array of 5 elements initially contains the elements as
3, 4, 5 , 16, 9
then the function should rearrange the content of the array as
9, 8, 15, 32, 27

Ans.
```
void replace(int Arr[ ], int size)
{
  for (int i=0; i<size; i++)
  if(Arr[i] % 2 != 0)
  Arr[i] * = 3;
  else
  Arr[i] * = 2;
}
```

(b) An array Array[20][15] is stored in the memory along the column with each element occupying 8 bytes. Find out the base address and address of the element Array[2][3] if the element Array[4][5] is stored at the address 1000.

Ans. Address of Array[i][j] along the column =
Base Address +W[(i-L1) + (j- L2) *M]
Where
W = size of each location in bytes = 8
L1 = Lower Bound of rows =0
L2 = Lower Bound of rows = 0

M = Number of rows per column =20

Address of Array[4][5] = Base Address + 8[(4-0)+ (5-0) * 20]

1000 = Base Address + 8[104]

Base Address =1000-8*104

=1000 – 832

= 168

Address of Arrays[2][3] = 168 + 8 [(2-0) + (3-0) * 20]

= 168 + 8* 62

= 168 + 496

= 664

(c) Write a function in c++ to delete a node containing Book's information from a dynamically allocated stack of Books implemented with the help of the following structure

```
struct Book
{
  int Bno ;
  char Bname[20];
  Book  *Next ;
  };
```

Ans.

```
Class stack
{
  public:
  book( )
{
  top=NULL; }
  void Push ( );
  void Pop( );
  void Display( );
  ~Book( );
};
  void stack :: Pop( )
{
        if (top != NULL)
{
        stack *temp;
        temp =Top;
        cout<<Top→Bno<<Top→ "deleted"<<endl;
        Top= Top → Next ;
        delete Temp;
}
        Else
        cout<< "stock Emplty" ;
  }
```

(d) Write a function in c++ which accepts a 2D array of integers and its size as arguments and displays the elements which lie on diagonals.

[Assuming the 2D Array to be square matrix with odd dimension ie. 3X3, 5X5, 7X7 etc…]

Example, if the array content is

5 4 3
6 7 8
1 2 9

Output through the function should be :

Diagonal One : 5 7 9
Diagonal two : 3 7 1

Ans.

```
void Diagonals(int Arr[][100], int size)
{ int Row, Col;
  cout<<"Diagonal One : ";
  for(Row =0;Row<Size;Row++)
        for(Col=0;Col<Size;Col++)
              if(Row = = Col)
                    cout<<Arr[Row][Col];
  cout<<"Diagonal Two : ";
  for(Row =0;Row<Size;Row++)
        for(Col=0;Col<Size;Col++)
              if(Row  + Col = = Size -1)
                    cout<<Arr[Row][Col];
  }
```

(e) Evaluate the following postfix notation of expression:

25 8 3 - / 6 * 10 +

Ans.

Operator Scanned	**Stack Content**
25	25
8	25, 8
3	25, 8, 3
-	25, 5
/	5
6	5, 6
*	30
10	30, 10
+	40

4.(a) Observe the program segment given below carefully and answer the questions that follows :

```
class PracFile
{       int Pracno;
        char PracName[20];
        int TimeTaken;
        int Marks;
```

```
public:
        //function to enter PracFile details
        void EnterPrac( );
        //function to display PracFile details
        void ShowPrac( );
        //function to return Time taken
        int RTime( ) { return TimeTaken;}
        //function to assign Marks
        void Assignmarks(int M)
        {
         Marks = M;
        };
void AllocateMarks( )
{       fstream File;
        File.open("Marks.Dat.,ios::binary|ios:in|ios:out);
        PracFile P;
        int Record = 0;
        while(File.read((char *)&P, sizeof(P)))
{       if (P.RTime( ) > 50)
        P.Assignment(0)
        else
        P.Assignment(10)
        _____//statement 1
        _____//statement 2
        Record++;
}
        File.Close( );
  }
```

If the function Allocate Marks( ) is supposed to allocate Marks for the records in the file MARKS.DAT based on their value of the member TimeTaken. Write C++ statements for the statement1 and statement2, where statement1 is required to position the file write pointer to an appropriate place in the file and statement2 is to perform the write operation with the modified record.

Ans.

Statement 1 :

```
File.seekp(Record * sizeof(P));
```

Or

```
File.seekp(Record * sizeof(PracFile));
```

Or

```
File.seekg(Record * sizeof(P));
```

Statement 2 :

```
File.write((char *)&P, sizeof(P));
```

Or
File.write((char *)&P, sizeof(PracFile));

(b) Write a function in C++ to print the count of the word is as an independent word in a text file DIALOGUE.TXT.

For example, if the content of the file DIALOGUE.TXT is
This is his book. Is this book good?
Then the output of the program should be 2.

Ans.
```
void countis( )
{       ifstream Fil;
        Fil.open("Dialogue.txt");
        char Word[50];
        int Count=0;
        while (!Fil.eof( ))
        {       Fil>>Word;
                if(strcmpi(Word,"is")==0)
                count++;
}
        cout<<count;
        Fil.close( );
}
```

(c) Given a binary file GAME.DAT, containing records of the following structure type

```
struct Game
{
  char GameName[20];
  char Participant[10][30];
};
```

Write a function in C++ that would read the contents from the file GAME.DAT and creates a file named BASKET.DAT copying only those records from GAME.DAT where the game name is "Basket Ball"

Ans :
```
void CopyBasket( )
{ Game G;
  ifstream fin;
  fin.open("GAME.DAT",ios::binary);
 ofstream fout;
 fout.open("BASKET.DAT",ios::binary);
 while(fin.read((char *)&G, Sizeof(G)))
 { if(strcmp(G.GameName,"Basket Ball")= =0)
  fout.write((char *)&G, sizeof(G));
 }
```

```
        fin.close( );
        fout.close( );
    }
```

5. (a) Differentiate between the terms primary key and alternate key.

Ans. All candidate keys, which are not the primary key of the table are called alternate keys.

(b) Consider the following tables Consignor, Consignee and Consignment. Write SQL commands for the statements (i) to (iv) and give the outputs for SQL queries (v) to (viii).

Sender

SenderID	SenderName	SenderAddress	City
ND01	R Jain	2, ABC Appts	New Delhi
MU02	H Sinha	12, Newtown	Mumbai
MU15	S Jha	27/A, Park Street	Mumbai
ND50	T Prasad	122 – K, SDA	New Delhi

Recipient

RecID	SenderID	RecName	RecAddress	RecCity
KO05	ND01	R Bajpayee	5, Central Avenue	Kolkata
ND08	MU02	S Mahajan	116, A Vihar	New Delhi
MU19	ND01	H Singh	2A, Andheri East	Mumbai
MU32	MU15	P K Swamy	B5, C S Terminus	Mumbai
ND48	ND50	S Tripathi	13, B1 D, Mayur Vihar	New Delhi

(i) To display the names of all Senders from Mumbai

Ans. SELECT sendername from Sender where sendercity='Mumbai';

(ii) To display the RecIC, Sendername, SenderAddress, RecName, RecAddress for every Recipient.

Ans. Select R.RecIC, S.Sendername, S.SenderAddress, R.RecName, R.RecAddress from Sender S, Recepient R where S.SenderID=R.SenderID;

(iii) To display Recipient details in ascending order of RecName

Ans. SELECT * from Recipent ORDER By RecName;

(iv) To display number of Recipients from each city

Ans. SELECT COUNT(*) from Recipient Group By RecCity;

(v) SELECT DISTINCT SenderCity from Sender;

Ans. SenderCity
Mumbai
New Delhi

(vi) SELECT A.SenderName, B.RecName
From Sender A, Recipient B
Where A.SenderID = B.SenderID AND B.RecCity ='Mumbai';

Ans.

A.SenderName	B.RecName
R Jain	H Singh

(vii) **SELECT RecName, RecAddress**
From Recipient
Where RecCity NOT IN ('Mumbai', 'Kolkata');

Ans.

RecName	RecAddress
S Mahajan	116, A Vihar
S Tripathi	13, BID, Mayur Vihar

(viii) **SELECT RecID, RecName**
FROM Recipent
Where SenderID='MU02' or SenderID='ND50';

Ans.

RecID	RecName
ND08	S Mahajan
ND48	S Tripathi

6.(a) State Distributive law and verify the same using truth table.

Ans. If X, Y, Z are Boolean Variables then

X.(Y + Z) = X.Y + X.Z **or** X+Y.Z = (X+Y).(X+Z)

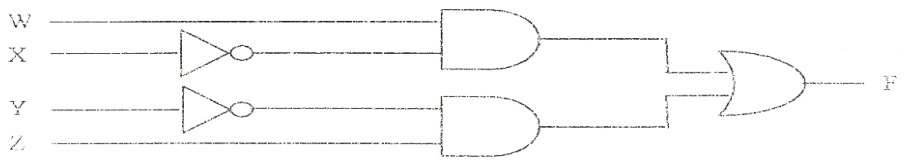
X	**Y**	**Z**	**Y+Z**	**X.(Y+Z)**	**X.Y**	**X.Z**	**X.Y+X.Z**
0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	1	1	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1

(b) Write the equivalent Canonical Sum of Product expression for the following Product of Sum Expression

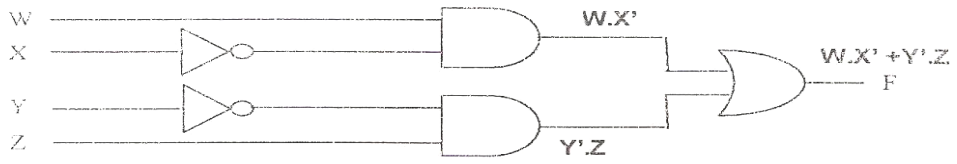
F(X, Y, Z) = ∏(1 , 3, 6, 7)

Ans. F(X, Y, Z) = ∑(0, 2, 4, 5)
=X'. Y'. Z' + X'.Y.Z' + X.Y'.Z' + X.Y'.Z

(c) Write the equivalent Boolean Expression for the following Logic circuit.



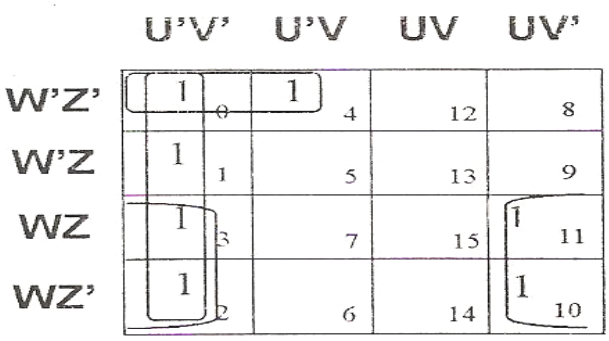
Ans:



F = W.X' + Y'.Z

(d) Reduce the following Boolean expression using K-Map
F(U,V,W,Z) = ∑ (0, 1, 2, 3, 4, 10, 11)

Ans.



F= U'.V' + W.V' + U'.W'.Z'

7.(a) What is the significance of Cyberlaw?

Ans. Cyberlaw helps prevent Cyber Crime, Hacking, Data Theft, Software Piracy and protects rights of Cyber Users.

(b) Expand the following terms with respect to networking :

(i) CDMA (ii) FTP
(iii) WLL (iv) HTML

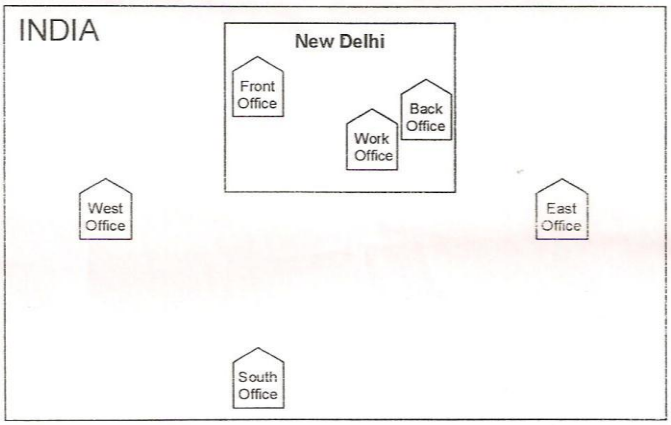
Ans. (i) Code Division Multiple Access
(ii) Wireless Local Loop
(iii) File Transfer Protocol
(iv) Hypertext Markup Language

(c) Which of the following unit measures the speed with which data can be transmitted from one node to another node of a network? Also, give the expansion of the suggested unit.

(i) Mbps (ii) KMph (iii) MGps

Ans. Mbps(Mega Bits Per Second)

(d) **"**Bhartiya Connectivity Association" is planning to spread their offices in four major cities in India to provide regional IT infrastructure support in the field of Education & Culture . The company has planned to setup their head office in New Delhi in three locations and have named their New Delhi offices as "Front office" ,"Back Office" and "Work Office". The company has three more regional offices as "South Office" , "East Office" and " West Office" located in other three major cities of India. A rough layout of the same as follows :



Approximate distances between these offices as per network survey team is as follows :

Place Form	**Place To**	**Distance**
Back Office	Front Office	10km
Back Office	Work Office	70 Meter
Back Office	East Office	1291km
Back Office	West Office	790 km
Back Office	South Office	1952 km

In continuation of the above , the company experts have planned to install the following number of computers in each of their offices :

Front Office	**100**
Work Office	20
East Office	50
West Office	50
South Office	50
Front Office	50

(1) **Suggest network types (out of LAN , MAN, WAN) for connecting each of the following set of their offices :**

- Back Office and Work Office
- Back Office and South Office

Ans. Back Office and Work Office -LAN
Back Office and South Office - WAN

(2) **Which device you will suggest to be procured by the company for connecting all the computers with in each of their offices out of the following devices?**

- Switch/Hub
- Modem
- Telephone

Ans. Switch / Hub

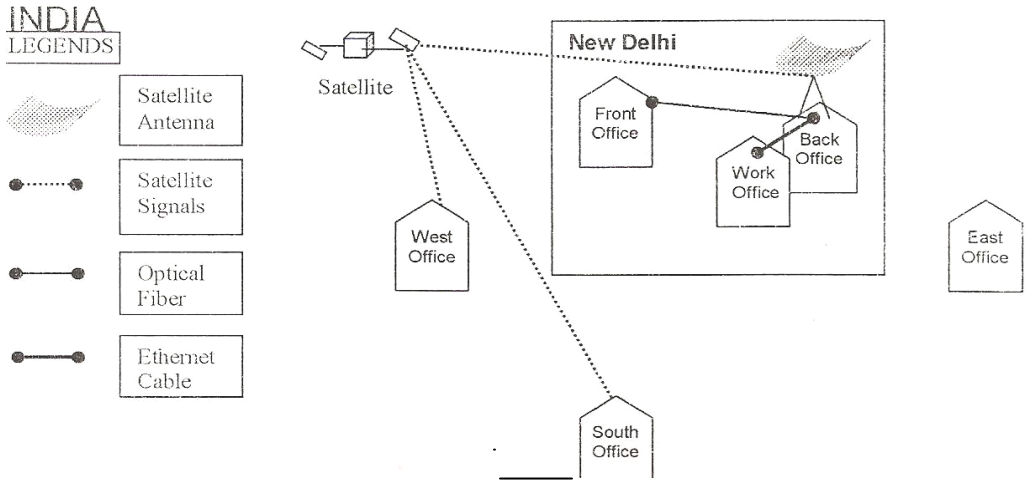
(3) Which of the following communication medium, you will suggest to be procured by the company for connecting their local offices in New Delhi for every effective and fast communication?

- Telephone Cable
- Optical Fiber
- Ethernet Cable

Ans. Optical Fiber

(4) Suggest a cable/wiring layout for connecting the company's local offices located in New Delhi. Also, suggest an effective method/technology for connecting the company's regional offices – "East Office", "West Office" and "South Office" with Offices located in New Delhi.

Ans.



Optical Fiber / Star Topology/ Radiowave etc.

CBSE BOARD EXAM 2007

Computer Science – 083

(**Solved Paper)**

Time Allowed : 3hours Maximum Marks 70

General Instructions

1. **All Questions are compulsory**
2. **Programming Language : C++**

1. (a) Differentiate between a Logical Error and Syntax Error. Also give suitable examples of each in C++. **2**

Ans :

Logical Error: Error occurred due to incorrect logic applied by the programmer.

Syntax Error: Error occurred due to not following the proper grammar/syntax of the language

OR

Error occurred due to violating rules of the programming language

Example:

```
//Program to find area and perimeter of rectangle
void main()
{
        int A,B,AR,P;
        A=10;
        B=20;
        AR=2*(A*B); //Logical Error – Wrong Formula
        P=2*(A+B);
        cout<<A<<P>>endl; //Syntax Error – Use of >> with cout
}
```

(b) Name the header file(s) that shall be needed for successful compilation of the following C++ code : **1**

```
void main( )
{
        char Text[40];
        strcpy(Text,"AISSCE");
        puts(Text);
}
```

Ans :

string.h
stdio.h

(c) Rewrite the following program after removing the syntactical error(s), if any. Underline each correction. **2**

```
#include <iostream.h>
```

```
const int Size 5;
void main()
{
        int Array[Size];
        Array = {50,40,30,20,10};
        for(Ctr=0; Ctr<Size; Ctr++)
        cout>>Array[Ctr];
}
```

Ans :

```
#include<iostream.h>
const int Size =5;
void main( )
{
        int Array[Size]={50,40,30,20,10};
        for(int Ctr=0;Ctr<Size;Ctr++)
        cout<<Array[Ctr];
}
```

(d) Find the output of the following program : **2**

```
#include<iostream.h>
void main()
{
        int Numbers[] = {2,4,8,10};
        int *ptr = Numbers;
        for (int C = 0; C<3; C++)
        {
                cout<< *ptr << "@";
                ptr++;
        }
        cout<<endl;
        for(C = 0; C<4; C++)
        {
                (*ptr)*=2;
                --ptr;
        }
        for(C = 0; C<4; C++)
                cout<< Numbers [C]<< "#";
        cout<<endl;
}
```

Ans :

```
2 @ 4 @ 8 @
4 # 8 # 16 # 20 #
```

(e) Find the output of the following program : **3**

```
#include<iostream.h>
void Indirect(int Temp=20)
{
        for (int 1=10; I<=Temp; I+=5)
        cout<<I<<" , " ;
        cout<<endl;
```

```
}
void Direct (int &Num)
{
        Num+=10;
        Indirect(Num);
}
void main()
{
        int Number=20;
        Direct(Number) ;
        Indirect();
        cout<< " Number=" <<Number<<endl ;
}
```

Ans:

```
10, 15, 20, 25, 30,
10, 15, 20,
Number=30
```

(f) In the following C++ program what is the expected value of Myscore from Options (i) to (iv) given below. Justify your answer. 2

```
#include<stdlib.h>
#include<iostream.h>
void main( )
{
        randomize();
        int Score[] = {25,20,34,56, 72, 63}, Myscore;
        Myscore = Score[2 + random(2)];
        cout<<Myscore<<endl;
}
```

(i) 25
(ii) 34
(iii) 20
(iv) None of the above

Ans :

(ii) 34

2. (a) Differentiate between Protected and Private members of a class in context of Inheritance using C++. 2

Ans :

Base Class	Access Specifier	Derived Class
Protected	Private	Private
	Protected	Protected
	Public	Protected
Private	Private	Not Accessible
	Protected	Not Accessible
	Public	Not Accessible

(b) Answer the questions (i) and (ii) after going through the following class: 2

```
class Science
{
        char Topic[20];
        int Weightage;
        public:
        Science ( ) //Function 1
        {
        strcpy (Topic, "Optics" );
        Weightage = 30;
        cout<<"Topic Activated";
        }
        ~Science( ) //Function 2
        {
                cout'<<"Topic Deactivated";
        }
```

(i) Name the specific features of class shown by Function 1 and Function 2 in the above example.

Ans :
Function 1: Constructor/ Default Constructor
Function 2: Destructor

(ii) How would Function 1 and Function 2 get executed ?

Ans:

Function 1 is executed or invoked automatically when an object of class Science is created.
Function 2 is invoked automatically when the scope of an object of class Science comes to an end.

OR

Example:

```
{
        Science s1;//Constructor is invoked
} // the destructor is invoke
```

(c) Define a class Travel in C++ with the description given below : 4

Private Members :
- T_Code of type string
- No_of_Adults of type integer
- No_of_Children of type integer
- Distance of type integer
- TotalFare of type float

Public Members :
- • A constructor to assign initial values as follows :
- T_Code with the word "NULL"
- No_of_Adults as 0
- No_of_Children as 0
- Distance as 0
- TotalFare as 0

• A function AssignFare( ) which calculates and assigns the value of the data member TotalFare as follows :
For **each** Adult
Fare (Rs) For Distance (Km)
500 >=1000
300 <1000 & >=500
200 <500
For **each** Child the above Fare will be 50% of the Fare mentioned in the above table.
For example :
If Distance is 750, No_of_Adults = 3 and No_of_Children = 2 Then TotalFare should be calculated as
No_of_Adults * 300 + No_of_Children * 150 i.e. 3 * 300 + 2 * 150 = 1200
• A function EnterTraveK) to input the values of the data members T_Code, No_of_Adults, No_of_Children and Distance; and invoke the AssignFare( ) **function.**
• A function ShowTraveK) which displays the content of all the data members for a Travel.

Ans :

```
class Travel
{
        char TCode[5]; //OR char *Tcode;
        int No_of_Adults;
        int No_of_Children;
        int Distance;
        float TotalFare;
        public:
                Travel();
                void AssignFare();
                void EnterTravel();
                void ShowTravel();
};
Travel::Travel()
{
        strcpy(TCode,"NULL");// OR TCode[0]='\0' OR strcpy(TCode,"\0")
        // OR TCode=NULL if TCode is declared as char pointer
        No_of_Adults = 0;
        No_of_Children = 0;
        Distance = 0;
        TotalFare = 0;
}
void Travel::AssignFare()
{
        if(Distance>=1000)
                TotalFare = 500*No_of_Adults+250*No_of_Children;
        else
                if (Distance >= 500)
                        TotalFare = 300*No_of_Adults+150*No_of_Children;
                else
                        TotalFare = 200*No_of_Adults+100*No_of_Children;
}
void Travel::EnterTravel()
{
        gets(TCode); // or cin >> TCode;
        cin>>No_of_Adults>>No_of_Children>>Distance;
        AssignFare();
```

```
}
void Travel::ShowTravel()
{
        cout<<TCode<<No_of_Adults<<No_of_Children <<Distance<<TotalFare<<endl;
}
```

(d) Answer the questions (i) to (iv) based on the following code : **4**

```
class Teacher
{
        char TNo[5], TName[20], DeptflO];
        int Workload;
        protected:
                float Salary;
                void AssignSal(float);
        public:
                Teacher( ) ;
                void TEntry( ) ;
                void TDisplay( );
};
class Student
{
        char Admno[10], SName[20], Stream[10];
        protected:
                int Attendance, TotMarks;
        public:
                Student( );
                void SEntry( );
                void SDisplay( );
};
class School : public Student, public Teacher
{
        char SCode[10], SchName[20];
        public:
                School ( ) ;
                void SchEntry( );
                void SchDisplay( );
};
```

(i) Which type of Inheritance is depicted by the above example ?

Ans : Multiple Inheritance

(ii) Identify the member functiion(s) that cannot be called directly from the objects of class School from the following :

TEntry( )
SDisplay( )
SchEntry( )

Ans :

None

(ii) Write name of all the member(s) accessible from member functions of class School.

Ans : Data Members: SCode, SchName, Attendance, TotMarks, Salary
Member Functions: SchDisplay(), SchEntry(), SEntry(), SDisplay( ),TEntry( ),

TDisplay( ),AssignSal( )

(iv) If class School was derived privately from class Teacher and privately from class Student, then, name the member function(s) that could be accessed through Objects of class School.

Ans : SchEntry( ), SchDisplay( ).

3. (a) 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. 4

Example : if an array of five elements initially contains the elements as

3, 4, 5, 16, 9

then the function should rearrange the content of the array as

6, 2, 10, 8, 18

Ans :

```
void Display(int NUM[],int N)
{
    for(int i=0;i<N;i=i+1)
    {
        if(NUM[i]%2==0)
            NUM[i]=NUM[i]/2;
        else
            NUM[i]=2*NUM[i];
    }
}
```

OR

```
void Display(int NUM[],int N)
{
    for(int i=0;i<N;i=i+1)
    NUM[i]=(NUM[i]%2!=0)?2*NUM[i]:NUM[i]/2;
}
```

(b) An array Arr[15][20] is stored in the memory along the row with each element occupying 4 bytes. Find out the Base Address and address of the element Arr[3][2], if the element Arr[5][2] is stored at the address 1500. 4

Ans :

Assuming LBR=LBC=0
S=4 bytes
Number of Rows(N)=15
Number of Columns(M)=20
LOC(Arr[I][J]). = B +((I-LBR)*M+(J-LBC))*S
LOC(Arr[5][2]). = B +((5-0)*20+(2-0))*4
1500. = B +(100+2)*4
B.= 1500-408
B.= 1092
LOC(Arr[3][2].= 1092+((3-0)*20+(2-0))*4
.. = 1092 + (62*4)
. .= 1092+248
. = 1340

OR

Assuming LBR=LBC=1
S=4 bytes

Number of Rows(N)=15
Number of Columns(M)=20
LOC(Arr[I][J]).= B +((I-LBR)*M+(J-LBC))*S
LOC(Arr[5][2]).= B +((5-1)*20+(2-1))*4
1500 .= B +(80+1)*4
B .= 1500-324
B .= 1176
LOC(Arr[3][2]).= 1176+((3-1)*20+(2-1))*4
. . = 1176 + (41*4)
.= 1176+164
. = 1340

(c) Write a function in C++ to delete a node containing customer's information, from a dynamically allocated Queue of Customers implemented with the help of the following structure : 4

```
struct Customer
{
        int CNo;
        char CName[20];
        Customer *Link;
};
```

Ans :

```
class QUEUE
{
        Customer *Rear,*Front;
        public:
                QUEUE( ) { Rear=NULL; Front=NULL;}
                void DELETE( );
                ~QUEUE( );
};
//Function definition DELETE()
void QUEUE::DELETE()
{
        if(Front==NULL) // OR if(!Front)
                cout<<"\n Queue Underflow\n";
        else
        {
                Customer *Temp;
                Temp=Front;
                cout<<Front->Cno<<":"<<Front->Cname <<"Deleted"<<endl;//To be ignored
                Front=Front->Link;
                delete Temp;
                if (Front==NULL)
                        Rear=NULL;
        }
}
```

(d) Write a function in C++ which accepts a 2D array of integers and its size as arguments and displays the elements of middle row and the elements of middle column. 2

[Assuming the 2D Array to be a square matrix with odd dimension i.e. 3×3, 5×5, 7×7 etc...]

Example, if the array content is
3 5 4
7 6 9
2 1 8
Output through the function should be :
Middle Row : 7 6 9
Middle Column : 5 6 1

Ans :

```
void Display(int A[][100],int N)
{
        cout<<"Middle Row:"<<endl;
        for(int i=0;i<N;i=i+1)
        cout<<A[N/2][i]<<" ";
        cout<<endl;
        cout<<"Middle Column:"<<endl;
        for(i=0;i<N;i=i+1)
                for(int i=0;i<N;i=i+1) – For Borland C++
        cout<<A[i][N/2]<<" ";
}
```

(e) Evaluate the following postfix notation of expression : 2
15 3 2 + / 7 + 2 *
Ans :
Evaluation of the given postfix expression is explained below

Operator Scanned	**Stack Content**
15	15
3	15, 3
2	15, 3, 2
+	15, 5
/	3
7	3, 7
+	10
2	10, 2
*	20

4. (a) Observe the program segment given below carefully, and answer the question that follows : 1

```
class Labrecord
{
        int Expno;
        char Experiment[20];
        char Checked;
        int Marks;
        public :
                //function to enter Experiment details
                void EnterExp( );
                //function to display Experiment details
                void ShowExp ( ) ;
                //function to return Expno
                char RChecked ( ) {return Checked;}
                //function to assign Marks
                void Assignmarks(int M)
```

```
        { Marks = M;}
};
void MpdifyMarks()
{
    fstream File;
    File.open("Marks.Dat",ios::binary|ios::in|ios::out);
    Labrecord L;
    int Rec = 0;
    while (File.read((char*)&L, sizeof(L)))
    {
        if(L.RChecked( )== ' N ' )
            L.Assignmarks(0)
        else
            L.Assignmarks(10)
        _____ //statement 1
        _____ //statement 2
        Rec ++ ;
    }
    File.close ();
}
```

If the funption ModifyMarks() is supposed to modify Marks for the records in the file MARKS.DAT based on their status of the member Checked (containing value either V or 'N'). Write C++ statements for the **statement 1** and **statement 2,** where, **statement 1** is required to position the file write pointer to an appropriate place in the file and **statement** 2 is to perform the write operation with the modified record.

Ans :

Statement 1:

```
File.seekg(-1*sizeof(L),ios::cur);
        OR
File.seekg(Rec*sizeof(L));
        OR
File.seekp(-1*sizeof(L),ios::cur);
        OR
File.seekp(Rec*sizeof(L));
        OR
```

Statement 2:

```
File.write((char *) &L,sizeof(L));
        OR
```

(b) Write a function in C++ to print the count of the word the as an independent word inatextfileSTORY.TXT. 2

For example, if the content of the file STORY.TXT is

There was a monkey in the zoo. The monkey was very naughty. Then the output of the program should be 2.

Ans :

```
void thewordCount()
{
    ifstream Fil("STORY.TXT");
    char String[20];
    int C=0;
    while(!Fil.eof())
    {
```

```
            Fil>>String;
            if(strcmpi(String,"the")==0)
            C=C+1;
        }
        cout<<C<<endl;
        Fil.close();
}
```

(c) Given a binary file SPORTS.DAT, containing records of the following structure type :

```
struct Sports
{
        char Event[20];
        char Participant[10][30];
};
```

(d) Write a function in C++ that would read contents from the file SPORTS.DAT and creates a file named ATHLETIC.DAT copying only those records from SPORTS.DAT where the event name is "Athletics". **3**

Ans :

```
void SP2AT()
{
        fstream IS,OA;
        Sports S;
        IS.open("SPORTS.DAT",ios::binary|ios::in);
        OA.open("ATHLETIC.DAT",ios::binary|ios::out);
        while(IS.read((char*) &S,sizeof(S)))
        {
                if(strcmp(S.Event,"Athletics")==0)
                        OA.write((char *)&S,sizeof(S));
        }
        IS.close();
        OA.close();
}
```

5. (a) What is the importance of a Primary Key in a table ? Explain with a suitable example. 2

Ans :

The Primary Key is an attribute/set of attributes that identifies a tuple/ row/ record uniquely.
Example:
Rollnumber in the table STUDENT

(b) Consider the following tables Consignor and Consignee. Write SQL commands for the statements (i) to (iv) and give outputs for SQL queries (v) to (viii). 6

TABLE : CONSIGNOR

CnorlD	*CnorName*	CnorAddress	City
ND01	R Singhal	24, ABC Enclave	New Delhi
ND02	Amit Kumar	123, Palm Avenue	New Delhi
MU15	R Kohli	5/A, South Street	Mumbai
MU50	S Kaur	27-K, Westend	Mumbai

TABLE : CONSIGNEE

CneelD	CnorlD	CneeName	CneeAddress	*CneeCity*
MU05	ND01	Rahul Kishore	5, Park Avenue	Mumbai
ND08	ND02	P Dhingra	16/J, Moore Enclave	New Delhi
KO19	MU15	A P Roy	2A, Central Avenue	Kolkata
MU32	ND02	S Mittal	P245, AB Colony	Mumbai
ND48	MU50	B P Jain	13, Block D, A Vihar	New Delhi

(i) To display the names of all Consignors from Mumbai.'

Ans :

```
SELECT CnorName FROM CONSIGNOR WHERE City='Mumbai';
```

(ii) To display the CneelD, CnorName, CnorAddress, CneeName, CneeAddress for every Consignee.

Ans :

```
SELECT B.CneeID, A.CnorName, A.CnorAddress, B.CneeName , B.CneeAddress
FROM Consignor A, Consignee B
WHERE A.CnorID=B.CnorID;
```

(iii) To display consignee details in ascending order of CneeName.

Ans :

```
SELECT * FROM CONSIGNEE ORDER BY CneeName;
```

(iv) To display number of consignors from each city,

Ans :

```
SELECT City,Count(CnorID) FROM CONSIGNOR Group By City;
```

(v) SELECT DISTINCT City FROM CONSIGNEE;

Ans :

```
DISTINCT CneeCity
Mumbai
New Delhi
Kolkata
```

(vi)
```
SELECT A.CnorName, B.CneeName
FROM Consignor A, Consignee B
WHERE A.CnorID = B.CnorlD AND B.CneeCity = 'Mumbai';
```

Ans :

A.CnorName	B.CneeName
R Singhal	Rahul Kishore
Amit Kumar	S Mittal

(vii) SELECT CneeName, CneeAddress

```
FROM Consignee
WHERE CneeCity NOT IN ('Mumbai', 'Kolkata');
```

Ans :

CneeName	CneeAddress
P Dhingra	16/J,Moore Enclave
B P Jain	13,Block D,A Vihar

(viii)
```
SELECT CneelD, CneeName
FROM Consignee
WHERE CnorID='MU15' OR CnorID='ND01';
```

Ans :

CneeID	CneeName
MU05	Rahul Kishore
KO19	A P Roy

6. (a) State De Morgan's Theorems and verify the same using truth table. 2

Ans :

If X, Y ∈ B

(X+Y)'=X'.Y'

X	Y	X'	Y'	X+Y	(X+Y)'	X'.Y'
0	0	1	1	0	1	1
0	1	1	0	1	0	0
1	0	0	1	1	0	0
1	1	0	0	1	0	0

(X.Y)'=X'+Y'

X	Y	X'	Y'	X.Y	(X.Y)'	X'+Y'
0	0	1	1	0	1	1
0	1	1	0	0	1	1
1	0	0	1	0	1	1
1	1	0	0	1	0	0

(b) Write the equivalent Canonical Product of Sum Expression for the following Sum of Product Expression

F(X, Y, Z) = Σ□(0, 2, 4, 5) 2

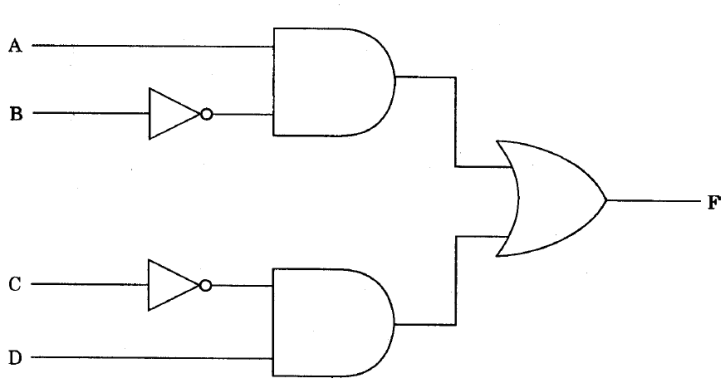
Ans :

F(X, Y, Z) = p (1, 3, 6, 7)

OR

F=(X+Y+Z')(X+Y'+Z')(X'+Y'+Z)(X'+Y'+Z')

(c) Write the equivalent Boolean expression for the following Logic Circuit. 2



Ans:

F=A.B'+C'.D

(d) Reduce the following Boolean expression using K-Map : **2**

F(A, B, C, D) = Π(5,6,7,8,9,12,13,14,15)

Ans:

	A'.B'	A'.B	A.B	A.B'
C'D'	1 (0)	1 (4)	12	8
C'.D	1 (1)	5	13	9
C.D	1 (3)	7	15	1 (11)
C.D'	1 (2)	6	14	1 (10)

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

7. (a) What is the significance of Cyber Law ? **1**

Ans :

Cyber law encompasses a wide variety of political and legal issues related to the Internet and other communications technology, including intellectual property, privacy, freedom of expression, and jurisdiction.

(b) Expand the following terms with respect to Networking : **2**

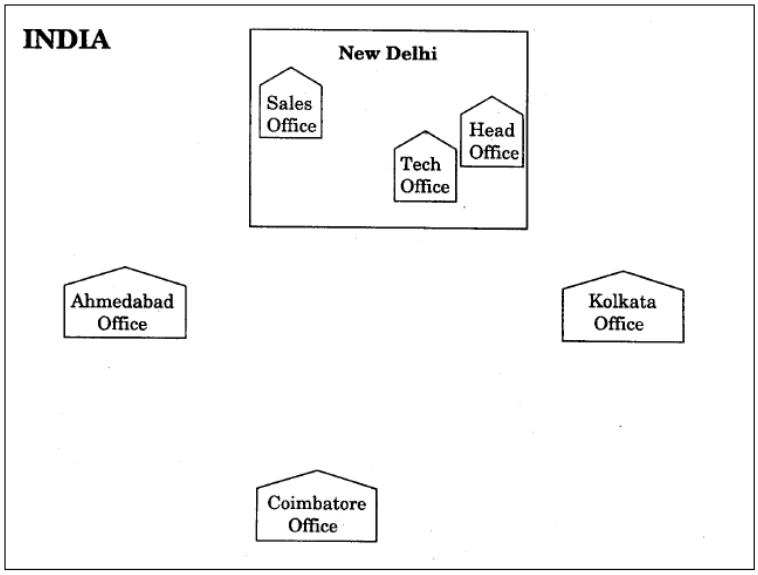
(i) XML - **eXtensible MarkUp Language**
(ii) WWW - **World Wide Web**
(iii) WLL - **Wireless in Local Loop**
(iv) TCP/IP - **Transmission Control Protocol/Internet Protocol**

(c) Which of the following units measures the speed with which data can be transmitted from one node to another node of a network ? Also, give the expansion of the suggested unit. **1**

(i) KMph
(ii) Mbps
(iii) MGps

Ans : (ii) Mbps

(d) "Hindustan Connecting World Association" is planning to start their offices in four major cities in India to provide regional IT infrastructure support in the field of Education & Culture. The company has planned to set up their head office in New Delhi in three locations and have named their New Delhi offices as "Sales Office", "Head Office" and "Tech Office". The company's regional offices are located at "Coimbatore", "Kolkata" and "Ahmedabad". A rough layout of the same is as follows :



Approximate distances between these offices as per network survey team is as follows :

Place From	Place To	Distance
Head Office	Sales Office	10 KM
Head Office	Tech Office	70 Meter
Head Office	Kolkata Office	1291 KM
Head Office	Ahmedabad Office	790 KM
Head Office	Coimbatore Office	1952 KM

In continuation of the above, the company experts have planned to install the following number of computers in each of their offices :

Head Office	100
Sales Office	20
Tech Office	50
Kolkata Office	50
Ahmedabad Office	50
Coimbatore Office	50

(i) Suggest network type (out of LAN, MAN, WAN) for connecting each of the following set of their offices :

- Head Office and Tech Office
- Head Office and Coimbatore Office

Ans : (i) · Head Office and Tech Office: **LAN**
· Head Office and Coimbatore Office: **WAN**

(ii) Which device will you suggest to be procured by the company for connecting all the computers within each of, their offices out of the following devices ?

• Modem
• Telephone
• Switch/Hub
Ans : Switch/Hub

(iii) Which of the following communication media, will you suggest to be procured by the company for connecting their local offices in New Delhi for very effective and fast communication ?
• Ethernet Cable
• Optical Fiber
• Telephone Cable
Ans : Optical fiber
(iv) Suggest a cable/wiring layout for connecting the company's local offices located in New Delhi. Also,, suggest an effective method/technology for connecting the company's regional offices at "Kolkata", "Coimbatore" and "Ahmedabad". 4

Ans : Optical Fiber/Star Topology
Wireless

CBSE BOARD EXAM 2006

Computer Science – 083

Time Allowed : 3hours Maximum Marks 70

General Instructions

3. **All Questions are compulsory**
4. **Programming Language : C++**

Q1. a. **Name the header file to which the following belong:** **(1)**

i. pow( )
ii. random( )

b. Illustrate the use of inline function in C++ with the help of an example. **(2)**

c. Rewrite the following program after removing the syntactical error(s), if any. Underline each correction. **(2)**

```
#include <iostream.h>
void main( )
{   struct movie
    {     char movie_name [20];
          char movie type;
    int ticket cost = 100;
}MOVIE;
    gets(movie_name);
    gets(movie_type);
}
```

d. Find the output of the following program: **(3)**

```
#include<iostream.h>
#include<string.h>
class student
{          char *name;
           int l ;
           public:
           student( ) {l=0; name=new char I1+1]; }
           student (char *s)
           {      I =strlen(s); name=new char[I+1];
                 strcpy (name,s);
           }
           void display( ) {cout<<name<<endl;}
           void manipulate(student & a, student & b)
           { I = a. I + b.I;
           delete name;
           name=new char[I+1];
           strcpy(name, a.name);
           strcat(name, b.name);
           }
};
```

```
void main( )
{     char * temp = “Jack”;
    student name1 (temp), name2(” Jill”), name3(”John”),S1,S2;
    S1 .manipulate (name1, name2);
    S2.manipulate (SI, name3);
    S1.display ( );
    S2.display ( );
}
```

e. Find the output of the following program:

```
#include<iostream.h>
void main()
{    long Number = 7583241;
    int First=0, Second=0;
    do
    {    int R=Number%10;
        if (R%2==0)
        First+=R;
        else
            Second+=R;
        Number /=10;
     } while (Number>O);
    cout<<First-Second;
}
```

f. What is a default constructor? How does it differ from destructor? **(2)**

Q2. What is “this” pointer? Give an example to illustrate the use of it in C++. (2)

a. Answer the questions (I) and (ii) after going through the following class:

```
class Exam
{    int year;
    public:
    Exam(int y) {year=y;}     //Constructor 1
    Exam(Exam & t);          ///Constructor 2
};
```

i. Create an object, such that it invokes Constructor I. **(1)**

ii. Write complete definition for Constructor 2. **(1)**

b. Define a class named HOUSING in C++ with the following descriptions: **(4)**

Private members

REG_NO integer(Ranges 10 — 1000)
NAME Array of characters(String)
TYPE Character
COST Float

Public Members

- Function Read_Data( ) to read an object of HOUSING type
- Function Display() to display the details of an object
- Function Draw Nos( ) to choose and display the details of 2 houses selected randomly from an array of 10 objects of type HOUSING Use random function to generate the registration nos. to match with REGNO from the array.

c. Answer the questions (i) to (iii) based on the following code:

```
class furniture
{
    char Type;
    char Model[10];
    public:
    furniture();
    void Read_fur_details( );
    void Disp_fur_detailsO;
};
    class sofa : public furniture
{
    int no_of_seats;
    float cost_of_sofa;
    public:
    void Read_sofa_details( );
    void Disp_sofa_details( );
};
    class office: private sofa
  {
    int no_of_pieces;
    char delivery_datel10l;
    public:
    void Read_office_details( );
    void Disp_office_details( );
  };
void main( )
{     office MyFurniture; }
```

. Mention the member names which are accessible by MyFurniture declared in main () function. **(1)**

i. What is the size of MyFurniture in bytes? **(1)**

ii. Mention the names of functions accessible from the member function Read_office_details () of class office. **(2)**

Q3.

a. Write a function in C++ which accepts an integer array and its size as arguments/parameters and assign the elements into a two dimensional array of integers in the following format: **(3)**

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

The resultant 2 D array is given below The resultant 2 D array is given below

b.

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

The resultant 2 D array is given below

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

If the array is 1, 2, 3

The resultant 2 D array is given below

1	0	0
1	2	0
1	2	3

c. An array MAT [20] [10] is stored in the memory along the row with each element occupying 4 bytes of memory. Find out the base address and the address of element MATE[10][5] if the location of MAT [3][7] is stored at the address 1000. **(4)**

d. Introduction class stack

```
{     int data [10];
      int top;
      public:
      stack( ) { top=-l }
      void push( ); //to push an element into the stack
      void pop( ); //to pop an element from the stack
      void Delete(int ITEM); //To delete all elements which are equal to ITEM
};
```

Complete the class with all function definitions. Use another stack to transfer data temporarily. **(4)**

e. Write a function in C++ to perform Insert operation in dynamically allocated Queue containing names of students. **(3)**

f. Write the equivalent infix expression for **(2)**
10,3, *, 7,1, *, 23, +

Q. 4.

a.

```
void main( )
{       char ch='A';
        fstream fileout(" data.dat", Ios::app);
        fileout<<ch;
        int p fileout.tellg( );
        cout<<p;
```

What is the output if the file content before the execution of the program is the string ? "ABC" (Note that" " are not part of the file) **(1)**

b. Write a function to count the number of blanks present in a text file named "PARA.TXT".(**2)**

c. Following is the structure of each record in a data file named "PRODUCT.DAT".

```
struct PRODUCT
{       char Prodact_Code[10];
        char Product_Descriptionil[10];
        int Stock;       };
```

Write a function in C++ to update the file with a new value of Stock. The Stock and the Product Code, whose Stock to be updated, are read during the execution of the program. **(3)**

Q. 5.

a. What are DDL and DML? **(2)**

b. Study the following tables FLIGHTS and FARES and write SQL commands for the questions (i) to (iv) and give outputs for SQL queries (v) to (vi).

TABLE : FLIGHTS

FL_NO	STARTING	ENDING	NO_FLIGHTS	NO STOPS
IC301	MUMBAI	DELHI	8	0
IC799	BANGALORE	DELHI	2	1
MC101	INDORE	MUMBAI	3	0
IC302	DELHI	MUMBAI	8	0
AM812	KANPUR	BANGALORE	3	1

IC899	MUMBAI	KOCHI	1	4
AM501	DELHI	TRIVANDRUM	1	5
MU499	MUMBAI	MADRAS	3	3
IC701	DELHI	AHMEDABAD	4	0

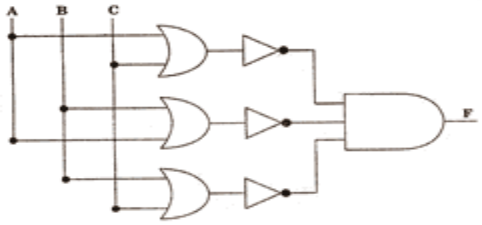
TABLE : FARES

FL_NO	AIRLINES	FARE	TAX%
1C701	Indian Airlines	6500	10
MU499	Sahara	9400	5
AM501	Jet Airways	13450	8
IC899	Indian Airlines	8300	4
1C302	Indian Airlines	4300	10<
1C799	Indian Airlines	10500	10
MC101	Deccan Airlines	3500	4

i. Display FL_NO and NO_FLIGHTS from "KANPUR" to "BANGALORE" from the table FLIGHTS.
ii. Arrange the contents of the table FLIGHTS in the ascending order of FL_NO.
iii. Display the FLNO and fare to be paid for the flights from DELHI to MUMBAI using the tables FLIGHTS and FARES, where the fare to be paid = FARE +FARE*TAX%/100.
iv. Display the minimum fare "Indian Airlines" is offering from the table FARES.
v. SELECT FL_NO, NO_FLIGHTS, AIRLINES from FLIGHTS, FARES where STARTING="DELHI" and FLIGHTS.FL_NO=FARES.FL_NO.
vi. SELECT count (distinct ENDING) from FLIGHTS. **(6)**

Q. 6.

a. State and verify Associative Law. **(2)**
b. Write the equivalent expression for the following logical circuit: **(2)**



c. Express P+Q 'R in POS form. **(1)**
d. Reduce the following Boolean expression using K-Map: **(3)**
F(P, Q, R, S) = π (0,3,5,6,7, 11, 12, 15)

Q. 7.

a. Name two transmission media for networking. **(1)**
b. Expand the following terms: **(2)**
i. XML ii. GSM iii. SMS iv. MAN
c. Differentiate between Hackers and Crackers. **(1)**

d. INDIAN PUBLIC SCHOOL in Darjeeling is setting up the network between its different wings. There are 4 wings named as SENIOR(S), JUNIOR(J), ADMIN(A) and HOSTEL(H). Distance between various wings are given below:

Wing A to Wing S	100m
Wing A to Wing J	200m
Wing A to Wing H	400m
Wing S to Wing J	300m
Wing S to Wing H	100m
Wing J to Wing H	450m

Number of Computers

Wing A	10
Wing S	200
Wing J	100
Wing H	50

i. Suggest a suitable Topology for networking the computer of all wings. **(1)**
ii. Name the wing where the Server is to be installed. Justify your answer. **(1)**
iii. Suggest the placement of Hub/Switch in the network. **(1)**
iv. Mention an economic technology to provide internet accessibility to all wings. **(1)**