KENDRIYA VIDYALAYA SANGATHAN

केंद्रीय विद्यालय संगठन

Kendriya Vidyalaya Sangathan

18, Institutional Area, Shaheed Jeet Singh Marg,
New Delhi - 110 602.

STUDY MATERIAL AND HOTS

Class XII (Computer Science 2010-11)

STUDY MATERIAL FOR CLASS XII COMPUTER SCIENCE

Chief Patron	**Shri Avinash Dikshit** Commissioner KVS, New Delhi
Patron	Joint Commissioner (Admn) KVS, New Delhi
Advisor	**Ms. Lakshmi Chari** Assistant Commissioner KVS,Ahmedabad Region
Guidance	**Sh. M.L.Gedam** Education Officer KVS, Ahmedabad Region **Sh.B. Rangasri** Education Officer KVS, Ahmedabad Region
Co-ordinator	**Sh. Venugopal** Principal, KV,Mehsana
Subject Coordinator	**Sh.Rahul Awasthi** PGT(Comp.Sc.),KV Mehsana **Sh. Rajesh Dwivedi** PGT(Comp.Sc.),KV Himmatnagar
Subject Contributors	**Sh.Naresh K Rana** PGT(Comp.Sc.), K.V.No.1, AFS, Bhuj **Sh. Yogesh Gupta** PGT(Comp.Sc.), K.V. Okha **Sh. Santosh K. Deval** PGT(Comp.Sc.), K.V.No.2, INF LINES, Jamnagar **Smt. Raksha Leuva** PGT(Comp.Sc.), K.V. Gandhinagar Cantt. **Sh. Om Prakesh Meena** PGT(Comp.Sc.), K.V. AFS Samana **Sh. S.S Mani** PGT(Comp.Sc.), K.V.No.2, ONGC, Surat **Sh. Abhivandan Mehta** PGT(Comp.Sc.), K.V.No. 3, AFII, Jamnagar

Salient features of this study material

- This study material is in the form of question bank comprising of both solved and unsolved questions from each chapter of the syllabus.
- It is a collection of a number of challenging questions based on HighOrder Thinking Skill of students. However, it shall prove to be a helping tool for all types of students.
- It aims at providing help to very high scorer students who may miss 100 out of 100 because of not being exposed to new type of questions, being used to only conventional types of question, and not paying attention towards the topics which are given in the reference books and syllabus of Computer Science as per CBSE guidelines (Forexample Relational Algebra in database concepts and extraordinary questions based on pointers in C++ programming).
- It contains guidelines, hints, and solutions for really challenging questions and topics.
- It contains a number of fresh/new questions(solved & unsolved), which shall increase the confidence level of the students when they solve them as perCBSE Guidelines
- Such questions shall draw the attention of both the students and the teachers, and the preparation to achieve 100% shall be boosted for all categories.

ALL THE BEST TO OUR DEAR STUDENTS**.....**

CBSE Mark Distribution for different Topics(Important Lessons)

SINo	Unit Name	Marks
1	UNIT 1 Programming in C++	30
2	UNIT 2 Data structures	14
3	UNIT 3 Database and SQL	08
4	UNIT 4 Boolean Logic	08
5	UNIT 5 Communication and open source concept	10
	Total Marks	**70**

Weightage to different topics/content units

S.No	Topics	Marks
1	Review of C++ covered in Class XI	12
2	Object Oriented Programming in C++	12
3	Data Structure & Pointers	14
4	Data File Handling in C++	06
5	Databases and SQL	08
6	Boolean Algebra	08
7	Communication and Open Source Concepts	10
	Total	70

Weightage to different forms of questions

S.No	Forms of Questions	Marks for each question	No. of Questions	Total Marks
1	Very Short Answer questions (VSA)	01	09	09
2	Short answer questions - Type I (SA I)	02	13	26
3	Short answer questions - Type II (SA II)	03	05	15
4	Long answer questions (LA)	04	05	20
		Total	32	70

Difficulty level of questions

S.No.	Estimated difficulty level	Percentage of marks
1	Easy	15%
2	Average	70%
3	Difficult	15%

STUDY MATERIAL

INDEX

STUDY MATERIAL'S THEORETICAL PART

UNIT 1 : PROGRAMMING IN C++

KEY POINTS:

Introduction to C++

- C++ is the successor of C language & developed by Bjarne Stroustrup at Bell Laboratories, New Jersey in 1979.

Tokens- smallest individual unit. Following are the tokens

- **Keyword**-Reserve word that can't be used as identifier
- **Identifies-**Names given to any variable, function, class, union etc.
- **Literals**-Value of specific data type
- **Variable**- memory block of certain size where value can be stored and changed.
- **Constant**- memory block where value can be stored once but can't changed later on
- **Operato**r – performs some action on data
    - Arithmetic(+,-,*,/,%)
    - Relational/comparison (<,>,<=,>=,==,!=).
    - Logical(AND(&&),OR(||),NOT(!).
    - Conditional (? :)
- **Precedence of operators:**

++(post increment),--(post decrement)	Highest
++(pre increment),--(pre decrement),sizeof !(not),-(unary),+unary plus)	
*(multiply), / (divide), %(modulus)	
+(add),-(subtract)	
<(less than),<=(less than or equal),>(greater than), >=(greater than or equal to)	
==(equal),!=(not equal)	
&& (logical AND)	
||(logical OR)	
?:(conditional expression)	
=(simple assignment) and other assignment operators(arithmetic assignment operator)	
, Comma operator	Low

Data type- A specifier to create memory block of some specific size and type

cout - Used to display value on monitor.

cin - To prompt value for specific variable.

comment-Used for better understanding of program statements and escaped by the compiler to compile . *e.g*. – single line (//) and multi line(/*….*/)

Control structure :

Sequence control statement(if)	**conditional statement (if else)**	**case control statement (switch case)**	**loop control statement (while ,do… while, for)**
Syntax	Syntax	Syntax	Syntax

if(condition)
{
statements;
}
	If(condition)
	{
	statements;
	}
	else
	{
	statements;
	}
		switch(expression)
		{
		case (expression1):
		[statements
		break;]
		case (expression2):
		[statements,
		break;]
		default:
		Statements;
		}
			while(condition)
			{
			statements;
			}
			do ….while loop
			do
			{
			statement;
			} while(condition);
			For loop
			for(initialization;condition;incr/decr)
			{
			statement;
			}

Nested loop -loop within loop.

exit()-to leave from the program.

break- exit from the current loop.

continue-to transfer control to loop control statement.

goto-program to jump a different location.

Character I/O Function:

- **get()-**input single character from standard input device.
- **put()-**output single character from standard output device
- **getch()-**read from keyboard
- **putch()-**write to screen.
- **getchar()-**return single character from a standard input device.
- **putchar()-**transmit single character to standard output device.
- **gets()-**used to read a string of character from the standard input file(stdin).
- **puts()-**used to copy a null-terminated string to standard output file (stdiout).

Standard C++ libraries

Header	Nome Purpose
stdio.h	File input and output
ctype.h	Character tests
string.h	String operations
math.h	Mathematical functions such as sin() and cos()
stdlib.h	Utility functions such as malloc() and rand()

Some More Useful header file.

fstream.h graphics.h conio.h ctype.h iomanip.h iostream.h
math.h stdlib.h stdio.h stream.h string.h time.h

Header files (Some important library functions):

stdio.h : fclose fcloseall feof fflush fgetchar fgets fileno fopen fprintf fputchar fputs fread fseek fwrite getc getchar gets

string.h : setmem stpcpy strcat strchr strcmp strcmpi strcpy strcspn stricmp strlen strlwr strncat strncmp strncmpi strncpy strnicmp strnset strrev strset strstr strupr

math.h : abs cos exp fabs floor fmod fmodl abs pow powl() sin sqrt tan

stdlib.h : abort abs atexit atof atoi atol calloc div exit free rand random randomize realloc

conio.h : cgets clrscr cprintf cputs cscanf getch getche gettext putch puttext textbackground textcolor

Some functions

- **isalpha(c)-**check whether the argument is alphabetic or not.
- **islower(c)-** check whether the argument is lowecase or not.
- **isupper(c)** - check whether the argument is upercase or not.
- **isdigit(c)-** check whether the argument is digit or not.
- **isalnum(c)-** check whether the argument is alphanumeric or not.
- **tolower()-**converts argument in lowercase if its argument is a letter.
- **toupper(c)-** converts argument in uppercase if its argument is a letter.
- **strcat()-** concatenates two string.
- **strcmp-**compare two string.
- **pow(x,y)-**return x raised to power y.
- **sqrt(x)-**return square root of x.
- **random-**return a random number between 0 and (num-1)
- **randomize-**intilizesthe random number generator with a random value.

Array- Collection of element of same type that are referred by a common name.

One Dimension array

- An array is a continuous memory location holding similar type of data in single row or single column

Two dimensional array

- A two diamensional array is a continuous memory location holding similar type of data in of both row sand columns (like a matrix structure).

Function-Self-contained block of code that does some specific task and return a value.

Function prototypes-Function declaration that specifies the return type and data type and no of arguments

syntax: return_type function_name(argument_type(s) argument_name(s));

Passing value to function-

- Passing by value
- Padding by address/reference

Function overloading

- Processing of using the same name for two or more function.

Function recursion

- Function that call itself either directly or indirectly.

Local variables

- Declared inside the function.

Global variables

- Declared before main function ().

Actual variables

- Variables associated with function name during function call.

Formal variables

- Variables which accept the actual variable inside the function.

Structure-Collection of logically related of different datatypes referenced under one name.

Nested structure

- A Structure definition within another structure.

typedef

- Used to define new data type name

#define Directives

- Use to define a constant number or function or to replace an instruction.

1 Marks questions

1) Name the header files that shall be needed for the following code:

```
void main( )
{
  char String[ ] = “String”;
  cout << setw(2)<<String;
}
```

2) Which C++ header file(s) will be essentially required to be include to run/execute the following C++ code: [CBSE-2010]

```
void main()
{
        int Rno=24; char name[ ]=”Alma Mater”;
        cout<<setw(10)<<Rno<<setw(20)<<name<<endl;
}
```

3) Name the header files that shall be needed for the following code:

```
void main( )
{
  char word[]=”Board Exam”;
  cout<<setw(20)<<word;
}
```

4) 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);
}
```

5) Name the header file(s) that shall be needed for successful compilation of the following C++ code.

```
void main( )
{
  char Text[40];
  strcpy(Text,”AISSCE”);
  puts(Text); }
```

6) Name the header file to which the following below:
(i) abs( ) (ii) isupper( )

7) Name the header file to which the following belong:
(i) pow ( ) (ii)random( )

8) Name the header files to which the following belong:
(i) abs( ) (ii) strcmp( )

9) Name the header files to which the following belong: [AI 2005]
(i) puts( ) (ii) isalnum( )

10) Write the names of the header files to which the following belong:
(i) gets( ) (ii) strcmp( ) (iii)abs( ) (iv)isalnum( )

11) Name the header file, to which the following built-in function belongs:
(i) strcmp( ) (ii)getc( )

12) Name the header files of C++ to which the following functions belong:
(i)get( ) (ii)open( ) (iii)abs( ) (iv)strcat( )

13) Name the header file to be included for the use of the following built in functions: (i)getc( ) (ii)strcat()

14) Name the header file, to which following built in function belong:

(i) isupper( ) (ii)setw() (iii)exp( ) (iv)strcmp( )

15) Why main( ) function is so special. Give two reasons?

16) Name the header file of C++ to which following functions belong.

(i)strcat( ) (ii) scanf( ) (iii) getchar( ) (iv)clrscr( )

17) Name the header files, to which the following built in functions belongs to:

(i)cos( )(ii)setw( )(iii)toupper( )(iv)strcpy( )

18) Name the header files, to which the following built in functions belongs to:

(i)cos( )(ii)setw( )(iii)toupper( )(iv)strcpy( )

19) Name the header file to, which following built-in functions belong:

(i) strcpy() *(ii)* isdigit() *(iii)* log() *(iv)* puts()

***20)* Name the header file to be included for the use of following built-in functions:**

(i) frexp() *(ii)* toupper() *[CBSE Sample Paper-2]*

21) Name the header flies of C++ to which the following functions belong:

(i) write() *(ii)* arc() *(iii)* open() *(iv)* strlen()

***22)* Name the header files of C++ to which the following functions belong:** *[AI2002]*

(i) get() (ii) open() (iii) abs() (iv) strcat()

***23)* Name the header files of C++ to which the following functions belong:** *[Comptt. 2002]*

(i) read() (ii) open() (iii) get() (iv) strcmp()

***24)* Name the header fLle,to which the following built-in functions belong:**

(i) strcpy 0 (ii) gets()

***25)* Name the header file, to which the following built-in functions belong:** *[AI2003]*

(i) strcmp() (ii) getc()

***26)* Write the names ofthe header flies to which the following belong:**

(i) sqrt() (ii) isalpha() (iii) puts() (iv) strcpy()

***27)* Write the names ofthe header flies to which the following belong:** *[AI 2004]*

(i) gets() (ii) strcmp() (iii) abs() (iv) isalnum()

***28)* Write the name of header flies to which the following belong:** *[Comptt 2004]*

(i) sqrt() (ii) strcpy() (iii) isalpha() (iv) open()

2 Marks questions:

---

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

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

2) Find the output of the following program:

```
#include<iostream.h>
void main( )
{ int U=10,V=20;
for(int I=1;I<=2;I++)
{ cout<<"[1]"<<U++<<"&"<<V  5 <<endl;
 cout<<"[2]"<<++V<<"&"<<U + 2 <<endl; } }
```

3) **Rewrite the following C++ program after removing the syntax error(s) if any. Underline each correction.** **[CBSE 2010]**

```
 include<iostream.h>
class FLIGHT
{
        Long FlightCode;
```

```
        Char Description[25];
public
        void addInfo()
        {
        cin>>FlightCode; gets(Description);
        }
        void showInfo()
        {
                cout<<FlightCode<<":"<<Description<<endl;
        }       };

void main( )
{       FLIGHT F;
        addInfo.F();
        showInfo.F; }
```

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

```
#include<iostream.h>
void main( )
{ One=10,Two=20;
Callme(One,Two);
Callme(Two); }
void Callme(int Arg1,int Arg2)
{ Arg1=Arg1+Arg2;
Count<<Arg1>>Arg2; }
```

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

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

Outputs:

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

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

```
#include<stdlib.h>
#include<iostream.h>
void main( )
{ randomize( );
char Area[ ][10]={" NORTH", "SOUTH", "EAST", "WEST"};
int ToGo;
for(int I=0; I<3;I++) {
ToGo=random(2) + 1;
cout<<Area[ToGo]<<":"; } }
```

Outputs:

(i) SOUTH : EAST : SOUTH : (ii) NORTH : SOUTH : EAST :
(iii) SOUTH : EAST : WEST : (iv) SOUTH : EAST : EAST :

7) 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 Numbers[Max];
Numbers = {20,50,10,30,40};
for(Loc=Max-1;Loc>=10;Loc--)
cout>>Numbers[Loc];          }
```

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

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

Ii) 25 (ii) 34 (iii) 20 (iv) None of the above.

9) Find the output of the following program

```
#include<iostream.h>
void main( )
{       long NUM=1234543;
        int F=0,S=0;
do
{       int R=NUM % 10;
        if (R %2 != 0)
                F += R;
        else
                S += R;
        NUM / = 10;
} while (NUM>0);
cout<<F-S;     }
```

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

```
#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<<end1;
else
cout<<Value + Multiple <<endl; }
```

11) Find the output of the following program

```
#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]+1;
                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 . 1;C<=Size/2;C++,R--)
        {       char Temp=Text[C];
                Text[C]=Text[R];
                Text[R]=Temp;       }
        cout<<Text<<endl;    }
```

12) The following code is from a game, which generate a set of 4 random numbers. Praful is playing this game, help him to identify the correct option(s) out of the four choices given below as the possible set of such numbers generated from the program code so that he wins the game. Justify your answer. [CBSE 2010]

```
#include<iostream.h>
#include<stdlib.h>
const int LOW=25;
void main()
{       randomize();
        int POINT=5, Number;
        for(int I=1;I<=4;I++)
        {       Number=LOW+random(POINT);
                cout<<Number<<":" <<endl;
                POINT--;       }       }
```

(i) 29:26:25:28: (ii)24:28:25:26:
(iii) 29:26:24:28; (iv)29:26:25:26:

13) Rewrite the following program after removing the syntactical errors (if any). Underline each correction.

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

3 Marks questions:[Hots]

1) Find the output of the following program; [Delhi 2008]

```
#include<iostream.h>
#include<ctype.h>
void main( )
{ char Text[ ] = "Mind@work!";
for(int I=0; Text[I]!='\0';I++)
{ if(!isalpha(Text[I]))
Text[I]="*";
else if(isupper(Text[I]))
```

```
        Text[I]=Text[I]+1;
        else
        Text[I] = Text[I+1]; }
        cout<<Text; }
```

2) Find the output of the following program:

```
#include<iostream.h>
#include<ctype.h>
void main( )
{       char Mystring[ ] = "what@OUTPUT!";
        for(int I=0; Mystring[I]!='\0';I++)
        { if(!isalpha(Mystring[I]))
        Mystring[I]='*';
        else if(isupper(Mystring[I]))
                Mystring[I]=Mystring[I]+1;
        else
                Mystring[I] =Mystring[I+1];
        } cout<<Mystring; }
```

3) Find the output of the following program.

```
#include<iostream.h>
void Withdef(int HisNum=30)
{       for(int I=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;          }
```

4) Find the output of the following program:

```
#include<iostream.h>
void Indirect(int Temp=20)
{       for(int I=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;          }
```

5) Find the output of the following program:

```
#include<iostream.h>
#include<ctype.h>
void Secret(char Msg[],int N);
void main( )
{       char SMS=" rEPorTmE";
        Secret(SMS,2);
        cout<<SMS<<endl;    }
void Secret(char Msg[],int N)
{       for(int c=10;Msg[]1='\0';c++)
```

```
if(c%2==0)
        Msg[c]= Msg[c]+N;
else if (isupper(Msg[c]))
Msg[c]=tolower(Msg[c]);
else
Msg[c]= Msg[c]-N;     }
```

6) Find the output of the following program: [CBSE 2010]

```
#include<iostream.h>
struct three_d
{       int x,y,z;      };
        void movein(three_d &t, int step=1)
        {       t.x+=step;
                t.y+=step;
                t.z+=step;      }
        void moveout(three_d &t, int step=1)
        {       t.x-=step;
                t.y+=step;
                t.z-=step;      }
        void main()
        {       three_d t1={10,20,5},t2={30,10,40};
                movein(t1);
                moveout(t2,5);
                cout<<t1.x<<","<<t1.y<<","<<t1.z<<endl;
                cout<<t2.x<<","<<t2.y<<","<<t2.z<<endl;
                movein(t2,10);
                cout<<t2.x<<","<<t2.y<<","<<t2.z<<endl;     }
```

7) Write the output of the following program:

```
#include<iostream.h>
int func(int &x,int y=10)
{ if(x%y==0) return ++x;else return y- -; }
void main( )
{       int p=20,q=23;
        q=func(p,q);
        cout<<p<<q<<endl;
        p=func(q);
        cout<<p<<q<<endl;
        q=func(p);
        cout<<p<<q<<endl;           }
```

8) Find the output of the following program. [CBSE 2010]

```
#include<iostream.h>
#include<ctype.h>
void Mycode(char Msg[],char ch)
{       for(int cnt=0;Msg[cnt]!='\0';cnt++)
                {       if(Msg[cnt]>='B'&& Msg[cnt]<='G')
                                Msg[cnt]=tolower(Msg[cnt]);
                        else
                                if(Msg[cnt]=='A'||Msg[cnt]=='a')
                                        Msg[cnt]=ch;
                                else
                        if(cnt%2==0)
                                Msg[cnt]=toupper(Msg[cnt]);
                        else
                                Msg[cnt]=Msg[cnt-1];
                        }       }
        void main()
```

```
{       char MyText[]="ApEACeDriVE";
        Mycode(MyText,'@');
        cout<<"NEW TEXT: "<<MyText<<" "<<endl;          }
```

9) Write a function in C++ to merge the contents of two sorted arrays A & B into third array C. Assuming array A and B are sorted in ascending order and the resultant array C is also required to be in ascending order.

10) Write a function in C++ to search for a BookNo from a binary file "BOOK.DAT", assuming the binary file is containing the objects of the following class. 3

```
class
{
int Bno;
char Title[20];
public:
int RBno(){return Bno;}
void Enter(){cin>>Bno;gets(Title);}
void Display(){cout<<Bno<<Title<<endl;}
};
```

Answer to Questions

1 Marks Answer

1) Ans) iomanip.h
iostream.h

2) Ans) iostream.h
iomanip.h

3) Ans: iostream.h
iomanip.h

4) Ans) stdio.h string.h

5) Ans: string.h, stdio.h

6) Ans) (i) abs( ) - math.h, stdlib.h, complex.h
(ii)isupper( ) - ctype.h

7)* Ans*:
(i) abs( ) - math.h, stdlib.h, complex.h
(ii) random( ) - stdlib.h

8) Ans) (i) abs( ) **-** stdlib.h, math.h, complex.h
strcmp( ) - string.h

9) Ans)
(i) puts( ) - stdio.h
(ii) isalnum( ) - ctype.h

10) Ans:
(i) gets( ) - stdio.h
(ii) strcmp( ) - string.h
(iii) abs( ) - math.h, stdlib.h,complex.h
(iv) isalnum( ) - ctype.h

11) Ans:
(i) strcmp( ) - string.h
(ii)getc( ) - stdio.h

12) Ans:
(i) get( ) - iostream.h
(ii) open( ) - fstream.h
(iii) abs( ) - math.h, stdlib.h
(iv) strcat( ) - string.h

13) Ans:

(i) getc( ) - stdio.h
(ii) strcat( ) - string.h

14) Ans)

(i) isupper( ) - ctype.h
(ii)setw( ) - iomanip.h
(iii)exp( ) - math.h
(iv)strcmp( ) - string.h

15) Ans) Execution of the program starts and ends at main( ). The main ( ) is the driver function of the program. If it is not present in a program, no execution can take place.

16) Ans: (i)strcat( ) - string.h
(ii)scanf( ) - stdio.h
(iii)getchar( ) - stdio.h
(iv)clrscr( ) - conio.h

17) Ans:

(i) cos( ) - math.h
(ii) setw( ) - iomanip.h
(iii) toupper( ) - ctype.h
(iv) strcpy( ) - string.h

18) Ans:

(i) cos( ) - math.h
(ii) setw( ) - iomanip.h
(iii) toupper( ) - ctype.h
(iv) strcpy( ) - string.h

19) Ans.

(i) string.h *(ii)* ctype.h *(iii)* math.h *(iv)* stdio.h

20) Ans. (i) math.h *(ii)* ctype.h

21) Ans. (i) fstream.h *(ii)* graphics.h *(iii)* fstream.h *(iv)* string.h

22) Ans.

(i) iostream.h (ii) fstream.h (iii) math.h *(iv)* string.h

23) Ans.

(i) fstream.h (ii) fstream.h (iii) iostream.h *(iv)* string.h

24) Ans.

(i) string.h (ii) stdio.h>

25) Ans.

(i) string.h (ii) stdio.h

26) Ans.

(i) math.h (ii) ctype.h (iii) math.h (iv) string.h

27) Ans. (i) stdio.h (ii) string.h (iii) math.h (iv) ctype.h

28) Ans. (i) math.h (ii) strcpy.h (iii) ctype.h (iv) fstream.h

2 marks Answers

1 Ans:

```
#include<iostream.h>
void test(int x,int y=20); //Prototype missing
void main( )
{ int F = 10, S = 20; //Data type missing
Text(F,S); //Comma to come instead of ;
Text(S);}
void Text(int x, int y)
{ x=x+y;
cout<<x<<y; //Output operator << required }
```

2 Ans: Output:

[1]10&15
[2]21&13

[1]11&16
[2]22&14

3Ans:
```
#include<iostream.h>
        class FLIGHT
        {
                long FlightCode;
                char Description[25];
        public:
                void addInfo()
                {
                cin>>FlightCode; gets(Description);
                }
                void showInfo()
                {
                        cout<<FlightCode<<":"<<Description<<endl;
                } };
        void main( )
        {       FLIGHT F;
                F.addInfo();
                F.showInfo;}
```

4Ans:
```
void Callme(int Arg1,int Arg2=20);
#include<iostream.h>
void main( )
{       int One=10,Two=20;
Callme(One,Two); //Given ; instead of ,
Callme(Two);  }
void Callme(int Arg1,int Arg2)
{       Arg1=Arg1+Arg2;
cout<<Arg1<<Arg2;}
```

5 Ans)

Since random(2) gives either 0 or 1, Fly value will be either 1 or 2.
(random(n) gives you any number between 0 to n-1) City[1] is .CHN.
City[2] is .KOL.
Since I value from 0 to 2 (ie<3), 3 iterations will takes place.
So the possible output consists 3 strings separated by :, each of them may be either .CHN. or .KOL..

So the possible output will be
(ii) CHN : KOL : CHN:
(iv) KOL :CHN : KOL:

6Ans) Since random(2) gives either 0 or 1, ToGo value will be either 1 or 2.
(random(n) gives you any number between 0 to n-1) Area[1] is .SOUTH.
Area[2] is .EAST.
Since I value from 0 to 2 (ie<3), 3 iterations will takes place. So the possible output consists 3 strings separated by :, each of them may be either .SOUTH. or .EAST..

So the possible output will be
(i) SOUTH : EAST : SOUTH :
(iv) SOUTH : EAST : EAST :

7 Ans)
```
#include<iostream.h>
const int Max=10;//Constant Variable .Max. must be
//initialized. Declaration Syntax Error
void main( )
{ int Numbers[Max]={20,50,10,30,40};
for(Loc=Max-1;Loc>=0;Loc--)
cout>>Numbers[Loc];}
```

8 Ans: Expected Output:
(iv) None of the above.

9 Ans: Output: 2

10)Ans:
```
#include<iostream.h>
const int Multiple=3;
void main( )
{ int 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; }
```

11Ans: Output:
```
cbse*eXAM*3116
6113*MXAe*esbc
```

12. 27:
27:
27:
26:

13.
```
#include <iostream.h>
class MEMBER
{
int Mno;
float Fees;
public:
void Register(){cin>>Mno>>Fees;}
void Display(){cout<<Mno<<" : "<<Fees<<endl;}
};
void main()
{
MEMBER M;
M.Register();
M.Display();
}
```

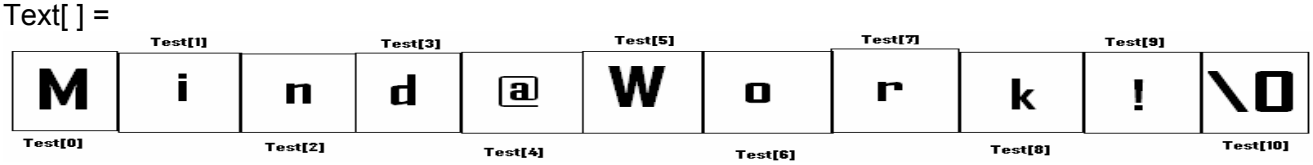
3 Marks answers

---

1)Ans:

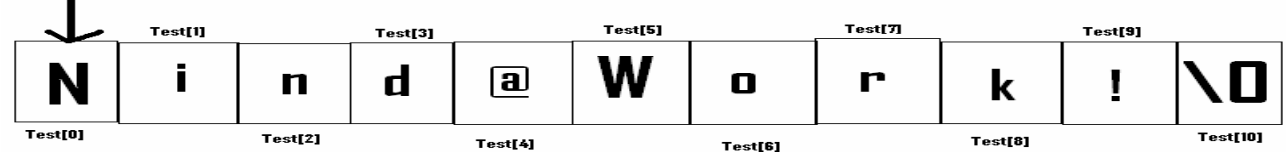
Solution:

Text[ ] =



When I=0

Since Text[0] is 'M', Upper Case Letter,
(isupper(Text[I]) will becomes true.
So Text[I] =Text[I]+1
So Text[0]=Text[0]+1
Text[0] =77(ASCII Value of M) + 1 = 78 =**N**(78 is ASCII Value of N)
Now the String Text[ ] =



When I=1

Since Text[1] is ‘i’, Which is a character, but which is not Upper case,

else part will be executed.

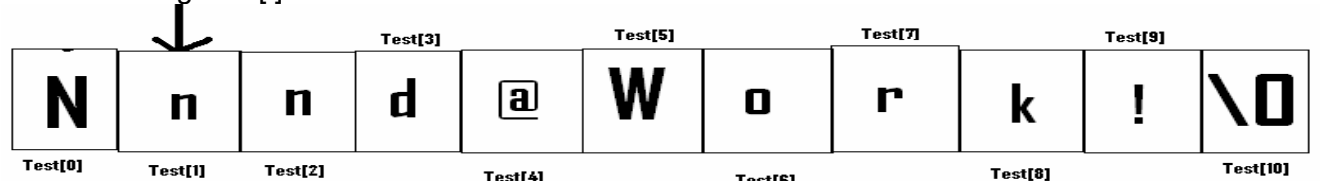
Ie Text[I]=Text[I+1]

Here Text[1]=Text[1+1]

=Text[2]

Ie ‘n’ will be stored in place of ‘I’

Now the String Text[ ] =



When I=2

Since Text[2] is ‘n’, Which is a character, but which is not Upper case, else part will be executed.

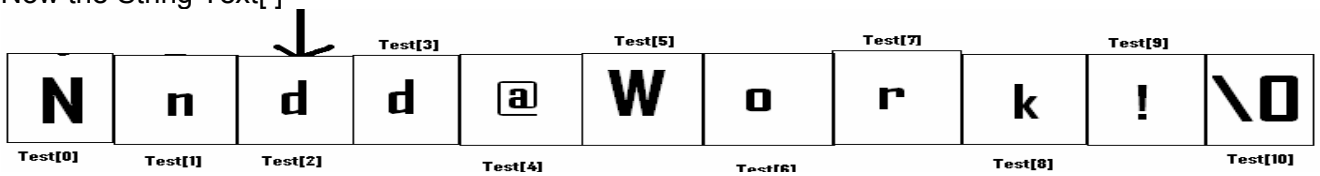
Ie Text[I]=Text[I+1]

Here Text[2]=Text[2+1]

=Text[3]

Ie ‘d’ will be stored in place of ‘n’

Now the String Text[ ] =



When I=3

Since Text[3] is ’d’, Which is a character, but which is not Upper case, else part will be executed.

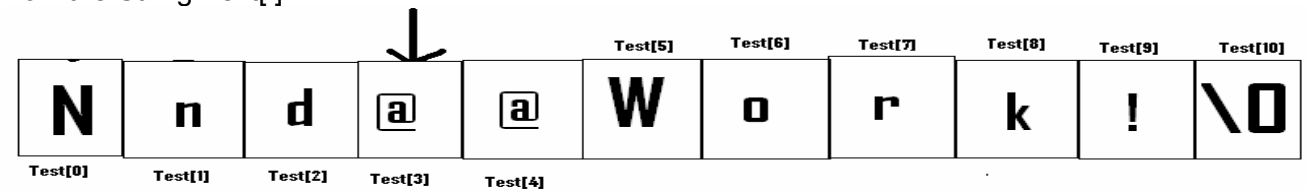
Ie Text[I]=Text[I+1]

Here Text[3]=Text[3+1]

=Text[4]

Ie ‘@’, will be stored in place of ,’d’

Now the String Text[ ] =



When I=4

Since Text[4] is ‘@’, Since which is not an alphabet, (!isalpha(Text[I])) will becomes true.

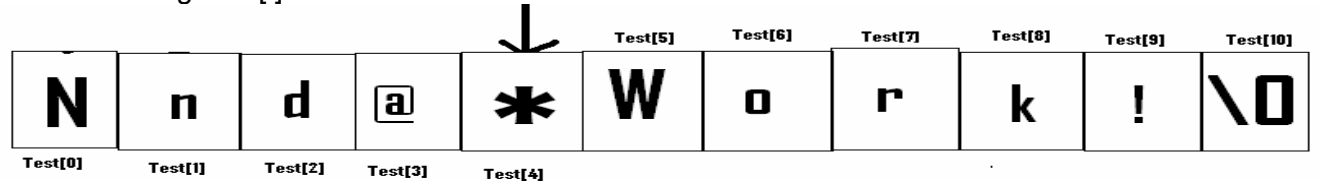
Ie **if(!isalpha(Text[I]))**

Text[I]=’*’;

Ie Text[4]=’*’

Ie ‘*’ will be stored in place of ’@’

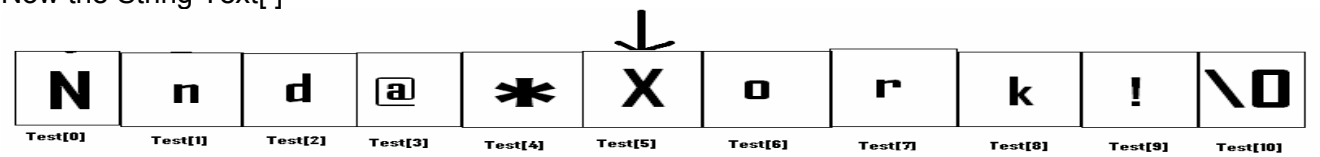
Now the String Text[ ] =



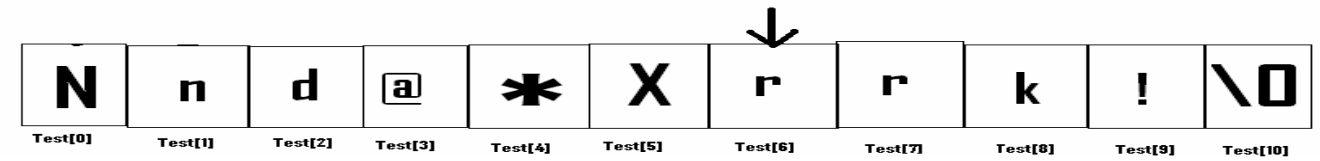
When I=5

Since Text[5] is ‘W’, Upper Case Letter,

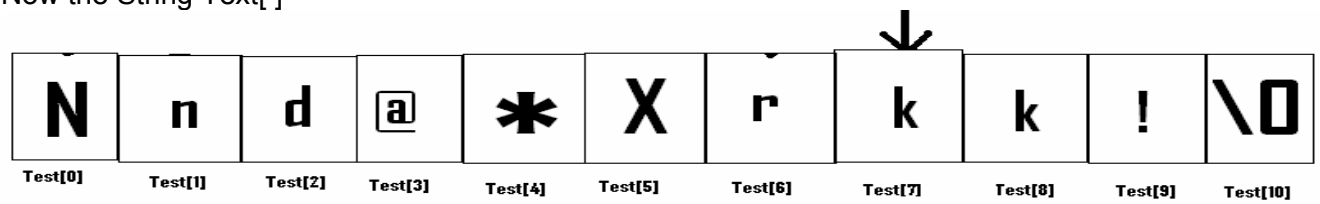
(isupper(Text[I]) will becomes true.
So Text[I] =Text[I]+1
So Text[5]=Text[5]+1
Text[5] =87(ASCII Value of W) + 1 = 88 =**X**(88 is ASCII Value of X)
Now the String Text[ ] =



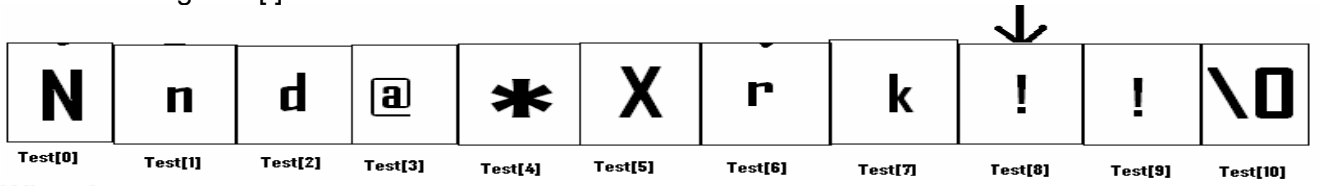
When I=6
Since Text[6] is 'o', Which is a character, but which is not Upper case, else part will be executed.
Ie Text[I]=Text[I+1]
Here Text[6]=Text[6+1]
=Text[7]
Ie 'r' will be stored in place of 'o'
Now the String Text[ ] =



When I=7
Since Text[7] is 'r', Which is a character, but which is not Upper case, else part will be executed.
Ie Text[I]=Text[I+1]
Here Text[7]=Text[7+1]
=Text[8]
Ie 'k' will be stored in place of 'r'.
Now the String Text[ ] =

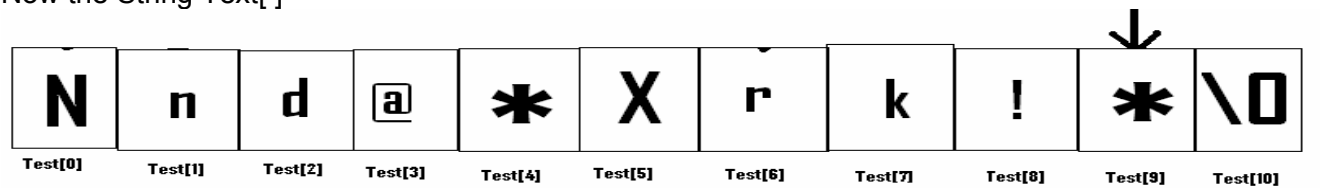


When I=8
Since Text[8] is 'k', Which is a character, but which is not Upper case, else part will be executed.
Ie Text[I]=Text[I+1]
Here Text[8]=Text[8+1]
=Text[9]
Ie '!' will be stored in place of 'k'
Now the String Text[ ] =



When I=9
Since Text[9] is '!', Since which is not an alphabet, (!isalpha(Text[I])) will becomes true.
Ie **if(!isalpha(Text[I]))**
Text[I]='*';
Ie Text[9]='*'

Ie '*' will be stored in place of '!'
Now the String Text[ ] =



Output: **Nnd@*Xrk!***

2) Ans:

Output: **hat@*PVUQVU***

3)Ans:

Output:
20,25,30,
20,25,30,
Number=30

4)Ans:

Output: 10,15,20,25,30,
10,15,20,
Number =30

5)Ans:

Output: teRmttoe

6) Ans:

Output:
11, 21 ,6
25 , 15, 35
35, 25, 45

7)Ans: Output:
2023
1023
1111

8)Ans: Output:
New Text=@@@@ccddIIe

9) Ans: Output:

```
void AddNSave(int A[ ],int B[ ],int C[ ],int N,int M, int &K) 3
{
int I=0,J=0;
K=0;
while (I<N && J<M)
if (A[I]<B[J])
C[K++]=A[I++];
else
if (A[I]>B[J])
C[K++]=B[J++];
else
{
C[K++]=A[I++];
J++;
}
for (;I<N;I++)
C[K++]=A[I];
for (;J<M;J++)
```

```
C[K++]=B[J];
}
```

10 Ans:Output:

```
void BookSearch()
{
fstream FIL;
FIL.open("BOOK.DAT",ios::binary|ios::in);
BOOK B;
int bn,Found=0;
cout<<"Enter Book No. to search…"; cin>>bn;
while (FIL.read((char*)&S,sizeof(S)))
if (FIL.RBno()==bn)
{
S.Display();
Found++;
}
if (Found==0) cout<<"Sorry! Book not found!!!"<<endl;
FIL.close();
   }
```

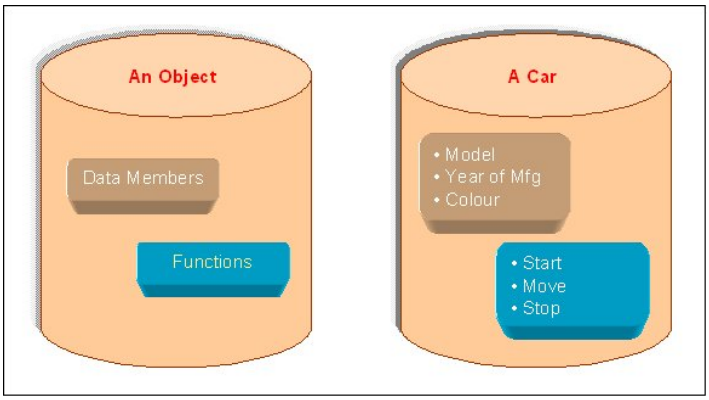
BASIC CONCEPTS OF OOPS

Some of the important object oriented features are namely:

- Objects
- Classes
- Inheritance
- Data Abstraction
- Data Encapsulation
- Polymorphism
- Overloading
- Reusability

Objects:

Object is the basic unit of object-oriented programming. Objects are identified by its unique name. An object represents a particular instance of a class..



- An Object is a collection of data members and associated member functions also known as methods.

Classes:

- Classes are data types based on which objects are created.
- Thus a Class represent a set of individual objects. Characteristics of an object are represented in a class as Properties. The actions that can be performed by objects becomes functions of the class and is referred to as Methods.

Inheritance:

- Inheritance is the process of forming a new class from an existing class or base class. The base class is also known as parent class or super class, The new class that is formed is called derived class.
- Derived class is also known as a child class or sub class. Inheritance helps in reducing the overall code size of the program, which is an important concept in object-oriented programming.

Data Abstraction:

- Data Abstraction increases the power of programming language by creating user defined data types.
- Data Abstraction also represents the needed information in the program without presenting the details.

Data Encapsulation:

- Data Encapsulation combines data and functions into a single unit called Class.
- Data Encapsulation enables the important concept of data hiding possible.

Polymorphism:

- Polymorphism allows routines to use variables of different types at different times.
- An operator or function can be given different meanings or functions.
- Polymorphism refers to a single function or multi-functioning operator performing in different ways.

Overloading:

- Overloading is one type of Polymorphism.
- It allows an object to have different meanings, depending on its context.
- When an exiting operator or function begins to operate on new data type, or class, it is understood to be overloaded.

Reusability:

- This term refers to the ability for multiple programmers to use the same written and debugged existing class of data.
- The programmer can incorporate new features to the existing class, further developing the application and allowing users to achieve increased performance.

IMPLEMENTATION OF OOPS IN C++

Constructors and Destructors

- Constructors and destructors are special member functions of classes that are used to construct and destroy class objects. Construction may involve memory allocation and initialization for objects.
- Destruction may involve cleanup and deallocation of memory for objects.

The following restrictions apply to constructors and destructors:

- Constructors and destructors do not have return types not even void nor can they return values.
- References and pointers cannot be used on constructors and destructors because their addresses cannot be taken.
- Constructors cannot be declared with the keyword virtual.
- Constructors and destructors cannot be declared static, const, or volatile.
- Unions cannot contain class objects that have constructors or destructors.
- The compiler automatically calls constructors when defining class objects and calls destructors when class objects go out of scope.
- . A constructor does not allocate memory for the class object its this pointer refers to, but may allocate storage for more objects than its class object refers to. If memory allocation is required for objects, constructors can explicitly call the new operator. During cleanup, a destructor may release objects allocated by the corresponding constructor. To release objects, use the delete operator.
- Derived classes do not inherit constructors or destructors from their base classes, but they do call the constructor and destructor of base classes.
- Constructors are also called when local or temporary class objects are created, and destructors are called when local or temporary objects go out of scope.
- You can call member functions from constructors or destructors.
- *constructor* is a member function with the same name as its class.
  For example:

```
class X {
        public:
                X();      // constructor for class X
        };
```

- *Destructors* are usually used to deallocate memory and do other cleanup for a class object and its class members when the object is destroyed.
- A destructor is called for a class object when that object passes out of scope or is explicitly deleted.
- A destructor is a member function with the same name as its class prefixed by a ~ (tilde).
  For example:

```
class X {
        public:

          X();      // Constructor for class X
         ~X();      // Destructor for class X

        };
```

Class members that are class types can have their own destructors. Both base and derived classes can have destructors, although destructors are not inherited. If a base class A or a member of A has a destructor, and a class derived from A does not declare a destructor, a default destructor is generated.
The default destructor calls the destructors of the base class and members of the derived class.
The destructors of base classes and members are called in the reverse order of the completion of their constructor:
The destructor for a class object is called before destructors for members and bases are called.
Destructors are implicitly called when an automatic object (a local object that has been declared auto or register, or not declared as static or extern) or temporary object passes out of scope. They are implicitly called at program termination for constructed external and static objects. Destructors are invoked when you use the delete operator for objects created with the new operator.

Default Constructors and Destructors

If you don't declare a constructor or a destructor, the compiler makes one for you. The default constructor and destructor take no arguments and do nothing.

What good is a constructor that does nothing? In part, it is a matter of form. All objects must be constructed and destructed, and these do-nothing functions are called at the right time.

Using constructors and destructors.

```
// Demonstrates declaration of a constructors and
// destructor for the Cat class

#include <iostream.h>          // for cout

class Cat                      // begin declaration of the class
{
 public:                       // begin public section
   Cat(int initialAge);        // constructor
   ~Cat();                     // destructor
   int GetAge();               // accessor function
   void SetAge(int age);       // accessor function
   void Meow();
 private:                      // begin private section
   int itsAge;                 // member variable
};

Cat::Cat(int initialAge)          // constructor definition  of Cat,
{
   itsAge = initialAge;
}

Cat::~Cat()                    // destroy the object of cat when it is no longer refered.
{
}

// GetAge, Public accessor function
// returns value of itsAge member
int Cat::GetAge()
{
   return itsAge;
}

// Definition of SetAge, public accessor function

void Cat::SetAge(int age)
{
                        // set member variable its age to  value passed in by parameter age
   itsAge = age;
}

// definition of Meow method
// returns: void
// parameters: None
// action: Prints "meow" to screen
void Cat::Meow()
{
   cout << "Meow.\n";
}
```

```
// create a cat, set its age, have it
// meow, tell us its age, then meow again.
int main()
{
  Cat Frisky(5);
  Frisky.Meow();
  cout << "Frisky is a cat who is " ;
  cout << Frisky.GetAge() << " years old.\n";
  Frisky.Meow();
  Frisky.SetAge(7);
  cout << "Now Frisky is " ;
  cout << Frisky.GetAge() << " years old.\n";
   return 0;
}
```

Output: Meow.
Frisky is a cat who is 5 years old.
Meow.
Now Frisky is 7 years old.

Copy Constructor

- A copy constructor is a special constructor in the C++ programming language used to create a new object as a copy of an existing object.
- Normally the compiler automatically creates a copy constructor for each class (known as a default copy constructor) but for special cases the programmer creates the copy constructor, known as a user-defined copy constructor. In such cases, the compiler does not create one.
- Copying of objects is achieved by the use of a copy constructor and a assignment operator. A copy constructor has as its first parameter a reference to its own class type. It can have more arguments, but the rest must have default values associated with them. The following would be valid copy constructors for class X:

```
X(const X& copyFromMe);
X(X& copyFromMe);
X(const X& copyFromMe, int = 10);
X(const X& copyFromMe, double = 1.0, int = 40);
```

The following cases may result in a call to a copy constructor:

- When an object is returned by value
- When an object is passed (to a function) by value as an argument
- When an object is thrown
- When an object is caught
- When an object is placed in a brace-enclosed initializer list

An object can be assigned value using one of the two techniques:

- Explicit assignment in an expression
- Initialization

Explicit assignment in an expression

```
Object A;
Object B;
A = B;      // translates as Object::operator=(const Object&), thus A.operator=(B) is called
          // (invoke simple copy, not copy constructor!)
```

Initialization

An object can be initialized by any one of the following ways.

a. **Through declaration**

Object B = A; // translates as Object::Object(const Object&) (invoke copy constructor)

b. Through function arguments

type function (Object a);

c. Through function return value

Object a = function();

The copy constructor is used only for initializations, and does not apply to assignments where the assignment operator is used instead.
The implicit copy constructor of a class calls base copy constructors and copies its members by means appropriate to their type. If it is a class type, the copy constructor is called. By using a user-defined copy constructor the programmer can define the behavior to be performed when an object is copied.

Examples

These examples illustrate how copy constructors work and why they are required sometimes.

Implicit copy constructor

Let us consider the following example.

```
//copy constructor
#include <iostream>
 class Person
{
public:
    int age;
 Person(int a)
    { age=a;}
};

int main()
{
    Person timmy(10);
    Person sally(15);
     Person timmy_clone = timmy;
     cout << timmy.age << " " << sally.age << " " << timmy_clone.age << endl;
     timmy.age = 23;
     cout << timmy.age << " " << sally.age << " " << timmy_clone.age << endl;
     return 0;
}
```

Output

```
10 15 10
23 15 10
```

As expected, *timmy* has been copied to the new object, *timmy_clone*. While *timmy's* age was changed, *timmy_clone's* age remained the same. This is because they are totally different objects.
The compiler has generated a copy constructor for us, and it could be written like this:

```
Person( Person& copy)
  { age=copy.age; }
```

INHERITANCE

- Inheritance is the process by which new classes called *derived* classes are created from existing classes called *base* classes.
- The derived classes have all the features of the base class and the programmer can choose to add new features specific to the newly created derived class.

Features or Advantages of Inheritance:

Reusability:

- Inheritance helps the code to be reused in many situations.
- The base class is defined and once it is compiled, it need not be reworked.
- Using the concept of inheritance, the programmer can create as many derived classes from the base class as needed while adding specific features to each derived class as needed**.**

Saves Time and Effort:

The above concept of reusability achieved by inheritance saves the programmer time and effort.
the main code written can be reused in various situations as needed.

Increases Program Structure which results in greater reliability.

Polymorphism

General Format for implementing the concept of Inheritance:

class derived_classname: access specifier baseclassname

For example, if the *base* class is *exforsys* and the derived class is sample it is specified as:

```
class sample: public exforsys
```

The above makes sample have access to both *public* and *protected* variables of base class *exforsys*.

Reminder about public, private and protected access specifiers:

- If a member or variables defined in a class is private, then they are accessible by members of the same class only and cannot be accessed from outside the class.
  .
- Public members and variables are accessible from outside the class.
  .
- Protected access specifier is a stage between private and public. If a member functions or variables defined in a class are protected, then they cannot be accessed from outside the class but can be accessed from the derived class.

Inheritance Example:

```
class exforsys
{
  public:
     exforsys(void) { x=0; }
   void f(int n1)
  {
  x= n1*5;
   }
 void output(void) { cout<<x; }
 private:
  int x;
 };
class sample: public exforsys
{
 public:
   sample(void) { s1=0; }
 void f1(int n1)
 {
 s1=n1*10;
  }
 void output(void)
 {
 exforsys::output();
 cout << s1;
 }
 private:
 int s1;          };

 int main(void)
 {    sample s;
 s.f(10);
 s.output();
 s.f1(20);
 s.output();
 }
```

The output of the above program is

50
200

Practice 1 :- Answer the questions after going through the following class.

```
class Exam
{
        char Subject[20] ;
        int Marks ;
public :
        Exam()                                  // Function 1
        {
strcpy(Subject, "Computer" ) ;
        Marks = 0 ;
        }
        Exam(char P[ ])                         // Function 2
        {
        strcpy(Subject, P) ;
        Marks=0  ;
        }
       Exam(int M)                              // Function 3
      {
              strcpy(Subject, "Computer") ;
              Marks = M ;
      }
       Exam(char P[ ], int M)                   // Function 4
      {
         strcpy(Subject, P) ;
Marks = M ;
      }   };
```

a) Which feature of the Object Oriented Programming is demonstrated using Function 1, Function2, Function 3 and Function 4 in the above class Exam?

Ans:- Function Overloading (Constructor overloading)

b) Write statements in C++ that would execute Function 3 and Function 4 of class Exam.

Ans:- Exam a(10); and Exam b("Comp", 10);

Practice 2: 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

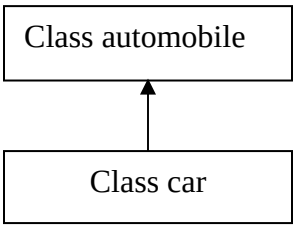
welcome obj (33, 'a9');	**valid**
welcome obj1(50, '9');	**valid**
welcome obj3();	**invalid**
obj1= welcome (45, 'T');	**valid**
obj3= welcome;	**invalid**

		Inheritance Mode		
		public	protected	private
Members in Base Class	public	public	protected	private
	protected	protected	protected	private
	private	X	X	X
		Members in derived class		

Types of Inheritance

1. Single class Inheritance:

- When class a gas inherited in class has known as base class and B class is know as derived class. Here only two classes have linked to each other.

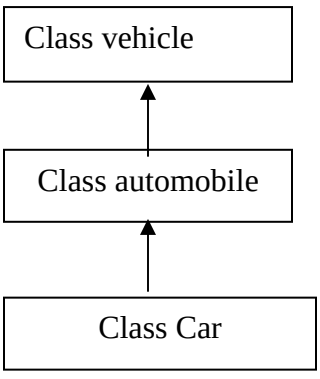


it is a Base class (super)

it is a sub class (derived)

2. Multilevel Inheritance:

- In this type of inheritance, there are number of level and it has used in that cases where we want to use all properties in number of levels according to the requirement



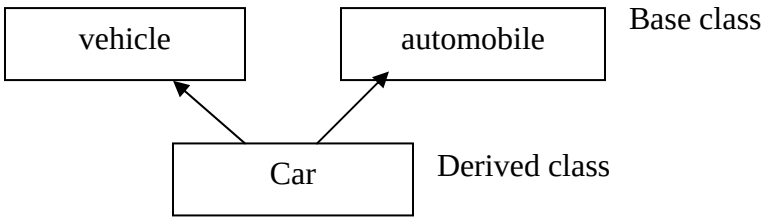
it is a Base class (super) of automobile

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

derived class(sub) of class automobile

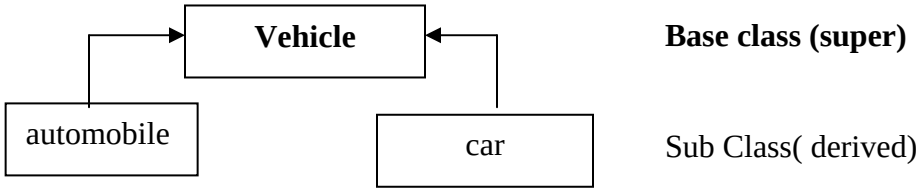
3. Multiple Inheritances:

- In this type of inheritance, number of classes has inherited in a single class.
- Where two or more classes are, know as base class and one is derive class.



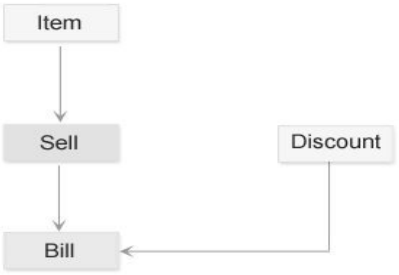
4. Hierarchical Inheritance:

- This type of inheritance helps us to create a baseless for number of classes and those numbers of classes can have further their branches of number of class.
-



5. Hybrid Inheritance:

- In this type of inheritance, we can have mixture of number of inheritances but this can generate an error of using same name function from no of classes, which will bother the compiler to how to use the functions.
- Therefore, it will generate errors in the program. This has known as ambiguity or duplicity.



Practice :- 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.

Practice :- Consider the following declarations and answer the questions given below:

```
#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

DATA FILE HANDLING IN C++

Key Points:

- Text file: A text file stores information in readable and printable form. Each line of text is terminated with an **EOL** (End of Line) character.

- Binary file: A binary file contains information in the non-readable form i.e. in the same format in which it is held in memory.

- Stream: A stream is a general term used to name flow of data. Different streams are used to represent different kinds of data flow.
- There are three file I/O classes used for file read / write operations.
  - **ifstream** - can be used for read operations.
  - **ofstream** - can be used for write operations.
  - **fstream** - can be used for both read & write operations.
- **fstream.h**:
- This header includes the definitions for the stream classes ifstream, ofstream and fstream. In C++ **file input output** facilities implemented through fstream.h header file.
- It contain predefines set of operation for handling file related input and output fstream.h class ties a file to the program for input and output operation.
- A file can be opened using:
  - By the constructor of the stream. This method is preferred when single file is used with the stream. (only for input / only for output)
  - By the open() function of the stream.
- **File modes:**
  - **ios::out** - It creates file in output mode and allows writing into the file.
  - **ios::in** - It creates file in input mode and permit reading from the file.
  - **ios::app** - To retain the previous contents of the file and to append to the end of existing file.
  - **ios::ate** - To place the file pointer at the end of the file, but you can write data any where in the file.
  - **ios::trunc** - It truncates the existing file (empties the file).
  - **ios::nocreate** - If file does not exist this file mode ensures that no file is created and open() fails.
  - **ios::noreplace** - If file does not exist, a new file gets created but if the file already exists, the open() fails.
  - **ios::binary –** Opens a file in binary.

eof():

- This function determines the end-of-file by returning true for end of file otherwise returning false.

open():If you want to manage multiple file with same stream use open().

Stream_object.open("Filename",(Filemode));

e.g., fstream fio;

fio.open("book.dat", ios::out | ios::in | ios::binary);

close(): This function terminates the connection between the file and stream associated with it.

Stream_object.close();

read(): The read() function reads a fixed number of bytes from the specified stream and puts them in the buffer.

Stream_object.read((char *)& Object, sizeof(Object));

write(): The write() function writes fixed number of bytes from a specific memory location to the specified stream.

Stream_object.write((char *)& Object, sizeof(Object));

Note:
Both functions take two arguments.
- The first is the address of variable, and the second is the length of that variable in bytes. The address of variable must be type cast to type char*(pointer to character type)
- The data written to a file using write( ) can only be read accurately using read( ).

get pointer: A get pointer indicates the position in the file at which the next input is to occur.

put pointer: It indicates the position in the file at which the next output is to be placed.

seekg(): It places the get pointer to the specified position in a stream.

seekp(): It places the put pointer to the specified position in a stream.

tellg(): This function returns the current position of the get pointer in a stream.
tellp(): This function returns the current position of the put pointer in a stream.

Steps To Process A File

- Determine the type of link required.
- Declare a stream for the desired types of link.
- Attach the desired file to the declared stream.
- Process the file.
- Close the file link with stream.

e.g
```
ifstream fin("book.txt");
char ch;
fin>>ch; //fin.get(ch);
cout<<ch;
```
→ **Open** file **book.txt**, **read** file and **display** the character from the file.

e.g,:
```
ofstream fout("book.txt");
char ch;
cin>>ch;
fout<<ch; // fout.put(ch);
```
→ **Create and open** file **book.txt**, write data into file from the variable.

General program structure used for operating a Text File

Text files in input mode:

Write a function in a C++ to count the number of lowercase alphabets present in a text file "BOOK.txt".

```
int countalpha()
{
        ifstream Fin("BOOK.txt");
        char ch;
        int count=0;
        while(!Fin.eof())
        {
                Fin.get(ch);
                if (islower(ch))
                        count++;
        }
        Fin.close();
        return count;
}
```

Function to calculate the average word size of a text file.

```
    void calculate()
    {
        fstream File;
        File.open("book.txt",ios::in);
        char a[20];
        char ch;
        int i=0,sum=0,n=0;
        while(File)
        {
            File.get(ch);
            a[i]=ch;
            i++;
            if((ch==' ') || ch(== '.'))
            {
                i --;
                sum=sum +i;
                i=0;
                N++;
            }
        }
    cout<<"average word size is "<<(sum/n);
    }
```

Write a program that prints a text file on the printer.

```
#include<iostream.h>
#include<fstream.h>
#include<process.h>
int main()
{
        char filename[13], ch;
        cout<<"enter the text file name :";
        cin.getline(filename,13);
        ifstream fin;
```

```
        fin.open(filename);
        if(!fin)
        {
        cerr<<"\nFile can't be opened !\n";
        exit(-1);
        }
        ofstream fout;
        fout.open("PRN");
        while(fin.get(ch)!=0)
        fout.put(ch);
        return 0;
}
```

Assume a text file "coordinate.txt" is already created. Using this file create a C++ function to count the number of words having first character capital.

Example:

Do less Thinking and pay more attention to your heart. Do Less Acquiring and pay more Attention to what you already have. Do Less Complaining and pay more Attention to giving. Do Less criticizing and pay more Attention to Complementing. Do less talking and pay more attention to SILENCE.

Output will be : Total words are 16

```
int countword()
{
        ifstream Fin("BOOK.txt");
        char ch[25];
        int count=0;
        while(!Fin.eof())
        {
                Fin>>ch;
                if (isupper(ch[0]))
                        count++;
        }
        Fin.close();
        return count;
}
```

Function to count number of lines from a text files (a line can have maximum 70 characters or ends at '.')

```
int countword()
{
        ifstream Fin("BOOK.txt");
        char ch[70];
        int count=0;
        if (!Fin)
        {
                cout<<"Error opening file!" ;
                exit(0);
        }
        while(1)
        {
                Fin.getline(ch,70,'.');
                if (Fin.eof())
                        break;
                count++;
        }
```

```
        Fin.close();
        return count;
}
```

A program to display the size of a file in bytes.

```
#include<iostream.h>
#include<fstream.h>
#include<process.h>
#include<conio.h>
int main()
{
        char filename[13];
        clrscr();
        cout<"Enter Filename:\n";
        cin.getline(filename,13);
        ifstream infile(filename);
        if(!infile)
        {
                cout>>"sorry ! Can not open "<<filename <<"file\n";
                exit(-1);
        }
        int no_bytes=0;
        char ch;
        while(cin.get(ch))
        {
                no_bytes ++;
}
        cout<<"File Size is"<<no_bytes<<"bytes\n";
return 0;
}
```

Text files in output mode:

C++ program, which initializes a string variable to the content "There is an island of opportunity in the middle of every difficulty." and output the string one character at a time to the disk file "OUT.TXT".

```
#include<fstream.h>
int main()
{
        ofstream fout("OUT.TXT");
        char *str = " There is an island of opportunity in the middle of every difficulty." ;
        int i=0;
        if(!fout)
        {
                cout<<"File cannot be opened ";
                return 0;
        }
        while (str[i]!='\0')
        {
                fout<<str[i];
                i++;
        }
        fout.close();
}
```

Exercise:

1. te a function in a C++ to count the number of uppercase alphabets present in a text file “BOOK.txt”
2. Write a function in a C++ to count the number of alphabets present in a text file “BOOK.txt”
3. Write a function in a C++ to count the number of digits present in a text file “BOOK.txt”
4. Write a function in a C++ to count the number of white spaces present in a text file “BOOK.txt”
5. Write a function in a C++ to count the number of vowels present in a text file “BOOK.txt”
6. Write a function in a C++ to count the average word size in a text file “BOOK.txt”
7. Write a function in C++ to print the count of the word “the” as an independent word in a text file STORY.TXT.
   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.

8. Assume a text file “Test.txt” is already created. Using this file, write a function to create three files “LOWER.TXT” which contains all the lowercase vowels and “UPPER.TXT” which contains all the uppercase vowels and “DIGIT.TXT” which contains all digits.
9. Create a function FileLowerShow() in c++ which take file name(text files)as a argument and display its all data into lower case
10. Write a function in C++ to count the number of lines present in a text file “Story.txt”.

HOTS FILE HANDLING

1. Write a function in a C++ to count the number of consonants present in a text file “Try.txt”
2. Write a function in a C++ to count the number of uppercase vowels present in a text file “Novel.txt”
3. Write a function in a C++ to display the sum of digits present in a text file “Fees.txt”.
4. Write a function in a C++ to display the product of digits present in a text file “Number.txt”.
5. Write a function in a C++ to find the largest digit present in a text file “Marks.txt”

General program structure used for operating a Binary File

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

```
struct student
{
char name[15];
float percent;
}

void main()
{
        clrscr();
        student s;
        strcpy(s.name,”rasha”);
        s.percent=89.50;
        ofstream fout;
        fout.open(“saving”, ios::out | ios:: binary);
        if(!fout)
        {
                cout<<“File can’t be opened”;
                break;
        }
        fout.write((char *) & s,sizeof(student));
        fout.close();
        ifstream fin;
        fin.open(“saving”,ios::in | ios:: binary);
```

```
        fin.read((char *) & s,sizeof(student));
        cout<<s.name;
        cout<<"\n has the percent: "<<s.percent;
        fin.close();
}
```

Function to add more objects belonging to class JOKE at the end of JOKES.DAT file.

```
void append()
{
        fstream afile;
        afile.open("JOKES.DAT", ios::binary | ios::app);
        JOKE LAUG;
        int n,i;
        cout<<"How many objects you want to add :";
        cin>>n;
        for (i=0;i<n;i++)
        {
                LAUG.Newjokeentry();
                afile.write((char *)& LAUG, sizeof (JOKE));
        }
        afile.close();
}
```

Write a function showfile() to read all the records present in an already exiting binary file SPEED.DAT and display them on the screen, also count the number of records present in the file. Assuming the class Vehicle as follows:

```
class vehicle
{
        char vehicletype[10];
        int no_of wheels;
        public:
        void getdetials()
        {
                gets(vehicletype);
                cin>>no_of_wheels;
        }
        void showdetails()]
        {
                cout<<"Vehicle Type"<<vehicletype;
                cout<<"Number of Wheels="<<no_of_wheels;
        }
};

void showfile()
{
         ifstream fin;
        fin.open("SPEED.DAT",ios::in | ios::binary);
        vehicle v1;
        int count=0;
        while (!fin.eof())
        {
                fin.read((char *)&v1,sizeof(v1));
                count++;
```

```
            v1.showdetails();
        }
        cout<<"Total number of records are "<<count;
}
```

Exercise:

1. Write a function in C++ to search for a BookNo from a binary file "BOOK.DAT", assuming the binary file is containing the objects of the following class.

```
class BOOK
{
int Bno;
char Title[20];
public:
int RBno(){return Bno;}
void Enter(){cin>>Bno;gets(Title);}
void Display(){cout<<Bno<<Title<<endl;}
};
```

2. Write a function in C++ to add new objects at the bottom of a binary file "STUDENT.DAT", assuming the binary file is containing the objects of the following class.

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

3. Observe the program segment carefully and answer the question that follows:

```
class item
{
int item_no;
char item_name[20];
public:
void enterDetail( );
void showDetail( );
int getItem_no( ){ return item_no;}
};
void modify(item x, int y )
{
fstream File;
File.open( "item.dat", ios::binary | ios::in | ios::out) ;
item i;
int recordsRead = 0, found = 0;
while(!found && File.read((char*) &i , sizeof (i)))
{
recordsRead++;
if(i . getItem_no( ) = = y )
{
_____//Missing statement
File.write((char*) &x , sizeof (x));
```

```
found = 1;
}
}
if(! found)
cout<<"Record for modification does not exist" ;
File.close() ;
}
```

If the function modify( ) is supposed to modify a record in the file " item.dat ", which item_no is y, with the values of item x passed as argument, write the appropriate statement for the missing statement using seekp( ) or seekg( ), whichever is needed, in the above code that would write the modified record at its proper place.

4. Observe the program segment carefully and answer the question that follows:

```
class item
{int item_no;
char item_name[20];
public:
void enterDetails( );
void showDetail( );
int getItem_no( ){ return item_no;}
};
void modify(item x )
{fstream File;
File.open( "item.dat", _____ ) ; //parameter missing
item i;
while(File .read((char*) & i , sizeof (i)))
{if(x . getItem_no( ) = = i . getItem_no( ))
{File.seekp(File.tellg( ) – sizeof(i));
File.write((char*) &x , sizeof (x));
}
else
File.write((char*) &i , sizeof (i));
}
File.close() ;
}
```

If the function modify( ) modifies a record in the file " item.dat " with the values of item x passed as argument, write the appropriate parameter for the missing parameter in the above code, so as to modify record at its proper place.

5. Observe the program segment carefully and answer the question that follows:

```
class item
{
int item_no;
char item_name[20];
public:
void enterDetail( );
void showDetail( );
int getItem_no( ){ return item_no;}
};
void modify(item x )
{
fstream File;
File.open( "item.dat", ios::binary|ios::in|ios::out ) ;
item i;
while(File .read((char*) & i , sizeof (i)))//Statement 1
```

```
{
if(x . getItem_no( ) = = i . getItem_no( ))
{
File.seekp(File.tellg( ) – sizeof(i));
File.write((char*) &x , sizeof (x));
}
}
File.close() ;
}
```

If the function modify( ) modifies a record in the file " item.dat" with the values of item x passed as argument, rewrite statement 1 in the above code using eof( ), so as to modify record at its proper place.

POINTERS

Key points to remember:

Pointer:

- A kind of memory variable that holds the address of other variable of same data type.
- **Reference operator (&)** As soon as we declare a variable, the amount of memory needed is assigned for it at a specific location in memory (its memory address).
- This reference to a variable can be obtained by preceding the identifier of a variable with an ampersand sign (&), known as reference operator, and which can be literally translated as "address of".

  For example:

  ```
  ted = &andy;
  ```

  Consider the following code fragment:

  ```
  andy = 25;
  fred = andy;
  ted = &andy;
  ```

The values contained in each variable after the execution of this, are shown in the following diagram:
First, we have assigned the value 25 to andy (a variable whose address in memory we have assumed to be 1776).

Dereference operator (*)

- a variable which stores a reference to another variable is called a pointer.
- Pointers are said to "point to" the variable whose reference they store.Using a pointer we can directly access the value stored in the variable which it points to.
- To do this, we simply have to precede the pointer's identifier with an asterisk (*), which acts as **dereference operator** and that can be literally translated to "value pointed by".
- Therefore, following with the values of the previous example, if we write:

  ```
  beth = *ted;
  ```

(that we could read as: "beth equal to value pointed by ted") beth would take the value 25, since ted is 1776, and the value pointed by 1776 is 25. You must clearly differentiate that the expression ted refers to the value 1776, while *ted (with an asterisk * preceding the identifier) refers to the value stored at address 1776, which in this case is 25. Notice the difference of including or not including the dereference operator (I have included an explanatory commentary of how each of
these two expressions could be read):

```
beth = ted; // beth equal to ted ( 1776 )
beth = *ted; // beth equal to value pointed by ted ( 25 )
```

Declaring variables of pointer types

- Due to the ability of a pointer to directly refer to the value that it points to, it becomes necessary to specify in its declaration which data type a pointer is going to point to. It is not the same thing to point to a char as to point to an int or a float.

- The declaration of pointers follows this format: type * name; where type is the data type of the value that the pointer is intended to point to. This type is not the type of the pointer itself! but the type of the data the pointer points to. For example:
    - int * number;
    - char * character;
    - float * greatnumber;

Now have a look at this code:

```
// my first pointer
#include <iostream>
int main ()
{
int firstvalue, secondvalue;
int * mypointer;
mypointer = &firstvalue;
*mypointer = 10;
mypointer = &secondvalue;
*mypointer = 20;
cout << "firstvalue is " << firstvalue << endl;
cout << "secondvalue is " << secondvalue << endl;
return 0;
}
```

```
firstvalue is 10
secondvalue is 20
```

```
// more pointers
#include <iostream>
int main ()
{
int firstvalue = 5, secondvalue = 15;
int * p1, * p2;
p1 = &firstvalue; // p1 = address of firstvalue
p2 = &secondvalue; // p2 = address of secondvalue
*p1 = 10; // value pointed by p1 = 10
*p2 = *p1; // value pointed by p2 = value pointed by
p1
p1 = p2; // p1 = p2 (value of pointer is copied)
*p1 = 20; // value pointed by p1 = 20
cout << "firstvalue is " << firstvalue << endl;
cout << "secondvalue is " << secondvalue << endl;
return 0;
}
```

```
firstvalue is 10
secondvalue is 20
```

Notice that there are expressions with pointers p1 and p2, both with and without dereference operator (*). The meaning of an expression using the dereference operator (*) is very different from one that does not: When this operator precedes the pointer name, the expression refers to the value being pointed, while when a pointer name appears without this operator, it refers to the value of the pointer itself (i.e. the address of what the pointer is pointing to).

Pointers and arrays

- The concept of array is very much bound to the one of pointer.
- In fact, the identifier of an array is equivalent to the address of its first element, as a pointer is equivalent to the address of the first element that it points to, so in fact they are the same concept.

For example, supposing these two declarations:
int numbers [20];
int * p;
The following assignment operation would be valid:
p = numbers;

- After that, p and numbers would be equivalent and would have the same properties.
- The only difference is that we could change the value of pointer p by another one, whereas numbers will always point to the first of the 20 elements of type int with which it was defined.
- Therefore, unlike p, which is an ordinary pointer, numbers is an array, and an array can be considered a *constant pointer*. Therefore, the following allocation would not be valid:
  numbers = p;
- Because numbers is an array, so it operates as a constant pointer, and we cannot assign values to constants.
- Due to the characteristics of variables, all expressions that include pointers in the following example are perfectly

valid:

```
// more pointers
#include <iostream>
int main ()
{
int numbers[5];
int * p;
p = numbers; *p = 10;
p++; *p = 20;
p = &numbers[2]; *p = 30;
p = numbers + 3; *p = 40;
p = numbers; *(p+4) = 50;
for (int n=0; n<5; n++)
cout << numbers[n] << ", ";
return 0;
}
```

10, 20, 30, 40, 50,

In arrays we used brackets ([]) to specify the index of an element of the array to which we wanted to refer. Well, these bracket sign operators [] are also a dereference operator known as *offset operator*. They dereference the variable they follow just as * does, but they also add the number between brackets to the address being dereferenced. For example:

```
a[5] = 0; // a [offset of 5] = 0
*(a+5) = 0; // pointed by (a+5) = 0
```

These two expressions are equivalent and valid both if a is a pointer or if a is an array.

Pointer initialization

- When declaring pointers we may want to explicitly specify which variable we want them to point to:

```
int number;
int *tommy = &number;
```

The behavior of this code is equivalent to:

```
int number;
int *tommy;
tommy = &number;
```

When a pointer initialization takes place we are always assigning the reference value to where the pointer points (tommy), never the value being pointed (*tommy).

As in the case of arrays, the compiler allows the special case that we want to initialize the content at which the pointer points with constants at the same moment the pointer is declared:

```
char * terry = "hello";
```
In this case, memory space is reserved to contain "hello" and then a pointer to the first character of this memory block is assigned to terry. If we imagine that "hello" is stored at the memory locations that start at addresses 1702, we can represent the previous declaration as: It is important to indicate that terry contains the value 1702, and not 'h' nor hello", although 1702 indeed is the address of both of these. For example, we can access the fifth element of the array with any of these two expression:

```
*(terry+4)
terry[4]
```

Both expressions have a value of 'o' (the fifth element of the array).

Pointer arithmetics

- To conduct arithmetical operations on pointers is a little different than to conduct them on regular integer data types.
- To begin with, only addition and subtraction operations are allowed to be conducted with them, the others make no sense in the world of pointers.
- But both addition and subtraction have a different behavior with pointers according to the size of the data type to which they point.

For example, let's assume that in a given compiler for a specific machine,

char takes 1 byte, short takes 2 bytes and long takes 4.

Suppose that we define three pointers in this compiler:

```
char *mychar;
short *myshort;
long *mylong;
```

and that we know that they point to memory locations 1000, 2000 and 3000 respectively.

So if we write:

```
mychar++;
myshort++;
mylong++;
```

mychar, as you may expect, would contain the value 1001. But not so obviously, myshort would contain the value 2002, and mylong would contain 3004, even though they have each been increased only once. The reason is that when adding one to a pointer we are making it to point to the following element of the same type with which it has been defined, and therefore the size in bytes of the type pointed is added to the pointer. This is applicable both when adding and subtracting any number to a pointer. It would happen exactly the same if we write:

```
mychar = mychar + 1;
myshort = myshort + 1;
mylong = mylong + 1;
```

Both the increase (++) and decrease (--) operators have greater operator precedence than the dereference operator (*), but both have a special behavior when used as suffix (the expression is evaluated with the value it had before being increased). Therefore, the following expression may lead to confusion:

```
*p++
```

Because ++ has greater precedence than *, this expression is equivalent to *(p++). Therefore, what it does is to increase the value of p (so it now points to the next element), but because ++ is used as postfix the whole expression is evaluated as the value pointed by the original reference (the address the pointer pointed to before being increased).

Notice the difference with:

```
(*p)++
```

Here, the expression would have been evaluated as the value pointed by p increased by one. The value of p (the pointer itself) would not be modified (what is being modified is what it is being pointed to by this pointer).

If we write:

```
*p++ = *q++;
```

Because ++ has a higher precedence than *, both p and q are increased, but because both increase operators (++) are used as postfix and not prefix, the value assigned to *p is *q before both p and q are increased. And then both are increased. It would be roughly equivalent to:

```
*p = *q;
```

```
++p;
++q;
```
Like always, I recommend you to use parentheses () in order to avoid unexpected results and to give more legibility to the code.

Pointers to pointers

- C++ allows the use of pointers that point to pointers, that these, in its turn, point to data (or even to other pointers). In order to do that, we only need to add an asterisk (*) for each level of reference in their declarations:

```
char a;
char * b;
char ** c;
a = 'z';
b = &a;
c = &b;
```

This, supposing the randomly chosen memory locations for each variable of 7230, 8092 and 10502, could be represented as: The value of each variable is written inside each cell; under the cells are their respective addresses in memory. The new thing in this example is variable c, which can be used in three different levels of indirection, each one of them would correspond to a different value:

• c has type char** and a value of 8092
• *c has type char* and a value of 7230
• **c has type char and a value of 'z'

Null pointer

- A null pointer is a regular pointer of any pointer type which has a special value that indicates that it is not pointing to any valid reference or memory address.
- This value is the result of type-casting the integer value zero to any pointer type.

```
int * p;
p = 0; // p has a null pointer value
```

- Do not confuse null pointers with void pointers.
- A null pointer is a value that any pointer may take to represent that it is pointing to "nowhere", while a void pointer is a special type of pointer that can point to somewhere without a specific type. One refers to the value stored in the pointer itself and the other to the type of data it points to.

Dynamic Memory

- Until now, in all our programs, we have only had as much memory available as we declared for our variables, having the size of all of them to be determined in the source code, before the execution of the program.
- But, what if we need a variable amount of memory that can only be determined during runtime? For example, in the case that we need some user input to determine the necessary amount of memory space.
- The answer is *dynamic memory*, for which C++ integrates the operators new and delete.

Operators new and new[]

- In order to request dynamic memory we use the operator new.
- new is followed by a data type specifier and -if a sequence of more than one element is required- the number of these within brackets []. It returns a pointer to the beginning of the new block of memory allocated. Its form is:

```
pointer = new type
pointer = new type [number_of_elements]
```

The first expression is used to allocate memory to contain one single element of type type. The second one is used to assign a block (an array) of elements of type type, where number_of_elements is an integer value representing the amount of these. For example:

```
int * bobby;
bobby = new int [5];
```

In this case, the system dynamically assigns space for five elements of type int and returns a pointer to the first element of the sequence, which is assigned to bobby. Therefore, now, bobby points to a valid block of memory with space for five elements of type int. The first element pointed by bobby can be accessed either with the expression bobby[0] or the expression *bobby. The second element can be accessed either with bobby[1] or *(bobby+1) and so on...

Operators delete and delete[ ]

- Since the necessity of dynamic memory is usually limited to specific moments within a program, once it is no longer needed it should be freed so that the memory becomes available again for other requests of dynamic memory.
- This is the purpose of the operator delete, whose format is:
    - delete pointer;
    - delete [] pointer;
- The first expression should be used to delete memory allocated for a single element, and the second one for memory allocated for arrays of elements.

Practice :- Rewrite the following codes after removing errors, if any, in the following snippet. Explain each error.

```
void main()
{
const int i = 20;
const int * const ptr = &i;
(*ptr)++;
int j = 15;
ptr = &j;
}
```

Practice: Give the output of the following program:

```
void main()
{
char *p = “School”;
char c;
c = ++ *p ++;
cout<<c;
}
```

Practice: Give the output of the following program:

```
void main()
{
int x [] = { 50, 40, 30, 20, 10}:
int *p, **q, *t;
p = x;
t = x + 1;
q = &t;
cout << *p << “,” << **q << “,” << *t++;
}
```

Practice: Give the output of the following program(Assume all necessary header files are included):

```
void main( )
{
```

```
char * x = “WorLd CuP”;
char c;
c = ++ *x ++;
cout<<c;
}
```

Practice: Give the output of the following program(Assume all necessary header files are included) :

```
void main( )
{
char *x = “WorLD CuP”;
char c;
c = ( *(x+1) ) ++ ;
cout<<c;
}
```

Practice. What will be the output of the program(Assume all necessary header files are included) :

```
void print (char * p )
{
p = "Comp";
cout<<"value is "<<p<<endl;
}
void main( )
{
char * x = "Class XII";
print(x);
cout<<"new value is "<<x<<endl;}
```

Practice: Give output of following code fragment:

```
char *msg = “a ProFile”;
for (int i = 0; i < strlen (msg); i++)
if (islower(msg[i]))
msg[i] = toupper (msg[i]);
else
if (isupper(msg[i]))
if( i % 2 != 0)
msg[i] = tolower (msg[i-1]);
else
msg[i--];
cout << msg << endl;
```

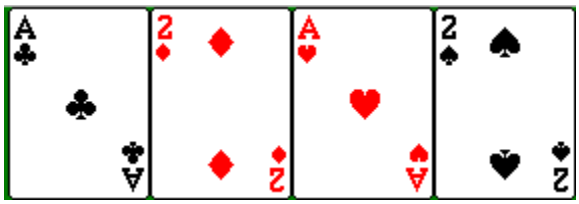
UNIT 2: DATA STRUCTURES

A **data structure** is a particular way of storing and organizing data in a computer so that it can be used efficiently.Different kinds of data structures are suited to different kinds of applications.
Types of data structure
There are two types of data structures
Linear data structure-Array,Linked list,Stack,Queue
Non-Linear data structue – Graph and Tree

Array-

An array of airplanes

An array of bugs



An array of cards

An array of characters

An array is a group of items that can be identified as similar because they are of the same nature.

Arrays come in two flavors: one dimensional and multi-dimensional arrays. Everyone of the pictures above represents a single dimensional array.

Declaring an Array

Like any other variable, the syntax of declaring an array is:

DataType ArrayName[dimension/order]

The array is first identified by its kind, which could be a char, an int, a float, etc; followed by its name that follows the C++ naming rules. The name is then followed by square brackets that specify the dimension of the array or its size.

Here are examples of declaring arrays:

```
int age[12];
float grade[100];
double angle[360];
```

int Age[12]; declares a group or array of 12 values, each one being an integer.

float Grade[100]; declares an array of 100 floating-point values.

double Angle[360]; declares an array of double-precision numbers. There are 360 of these items in the group.

Initializing an Array

Just like any variable can be initialized, an array also can be initialized. To accomplish this, for a one-dimensional array, the syntax used is:

DataType ArrayName[dimension] = { element1, element2, …, elementn};

Here are examples of declaring an initializing arrays:

```
int number[12] = {18, 42, 25, 12, 34, 15, 63, 72, 92, 26, 26, 12};
double distance[5] = {44.14, 720.52, 96.08, 468.78, 6.28};
```

Processing the Elements of an Array

```
#include <iostream>
int main()
{
	double distance[] = {44.14, 720.52, 96.08, 468.78, 6.28};
	cout << "2nd member = " << distance[1] << endl;
	cout << "5th member = " << distance[4] << endl;
	return 0;
}
```

This would produce:

```
2nd member = 720.52
5th member = 6.28
```

Operations on Arrays

```
#include <iostream>
int main()
{
	// We know that we need a constant number of elements
	const int max = 10;
	int number[max];
	// We will calculate their sum
	int sum = 0;
	cout << "Please type 10 integers.\n";
	for( int i = 0; i < max; i++ )
	{
		cout << "Number " << i + 1 << ": ";
		cin >> number[i];
		sum += number[i];
	}
	cout << "\n\nThe sum of these numbers is " << Sum << "\n\n";
	return 0;
}
```

Arrays and Functions

- An array can be passed to a function as argument.
- An array can also be returned by a function. To declare and define that a function takes an array as argument, declare the function as you would do for any regular function and, in its parentheses, specify that the argument is an array.Here is an example:

```
#include <iostream>
void DisplayTheArray(double member[5]);
int main()
{
	const int numberOfItems = 5;
```

```
	double distance[numberOfItems] = {44.14, 720.52, 96.08, 468.78, 6.28};
	return 0;
}
void DisplayTheArray(double member[5])
{
	for(int i = 0; i < 5; ++i)
		cout << "\nDistance " << i + 1 << ": " << member[i];
	cout << endl;
}
```

You don't have to specify the dimension of the array. This means that you can leave the square brackets empty:

```
#include <iostream>
void DisplayTheArray(double member[]);
int main()
{
	const int NumberOfItems = 5;
	double distance[NumberOfItems] = {44.14, 720.52, 96.08, 468.78, 6.28};
	return 0;
}

void DisplayTheArray(double member[ ])
{
	for(int i = 0; i < 5; ++i)
		cout << "\nDistance " << i + 1 << ": " << member[i];
	cout << endl;
}
```

Practice :

```
#include <iostream>

void DisplayTheArray(double member[])
{
	for(int i = 0; i < 5; ++i)
		cout << "\nDistance " << i + 1 << ": " << member[i];
	cout << endl;
}

int main()
{
	const int numberOfItems = 5;
	double distance[numberOfItems] = {44.14, 720.52, 96.08, 468.78, 6.28};
	cout << "Members of the array";
	DisplayTheArray(distance);
	return 0;
}
```

This would produce:

```
Members of the array
Distance 1: 44.14
Distance 2: 720.52
Distance 3: 96.08
Distance 4: 468.78
Distance 5: 6.28
```

Practice :-#include <iostream>

```
void DisplayTheArray(double member[])
{
        for(int i = 0; i < 5; ++i)
                cout << "\nDistance " << i + 1 << ": " << member[i];
        cout << endl;
}

int main()
{
        const int NumberOfItems = 5;
        double distance[NumberOfItems] = {44.14, 720.52, 96.08, 468.78, 6.28};
        cout << "Members of the array";
        DisplayTheArray(distance[3]);
        return 0;
}
```

Practice:- #include <iostream>

```
void DisplayTheArray(double mbr[ ], int count);
int main()
{
        double distance[] = {44.14, 720.52, 96.08, 468.78, 6.28, 68.04, 364.55, 6234.12};
        // Processing 5 members of the array
        cout << "Members of the array";
        DisplayTheArray(distance, 5);
        // Processing all members of the array
        int sizeOfArray = sizeof(Distance)/sizeof(double);
        cout << "\nMembers of the array";
        DisplayTheArray(distance, sizeOfArray);
        return 0;
}

void DisplayTheArray(double member[], int counter)
{
        for(int i = 0; i < counter; ++i)
                cout << "\nDistance " << i + 1 << ": " << member[i];
        cout << endl;
}
```

This would produce:

```
Members of the array
Distance 1: 44.14
Distance 2: 720.52
Distance 3: 96.08
Distance 4: 468.78
Distance 5: 6.28
Members of the array
```

Searching

Linear search: Linear search or **sequential search** is a method for finding a particular value in a list, that consists of checking every one of its elements, one at a time and in sequence, until the desired one is found. function for linear search

```
#include <iostream>
int LinearSearch(int Array[], const int Size, const int ValToSearch)
{	bool NotFound = true;
	int i = 0;
	while(i < Size && NotFound)
	{
		if(ValToSearch != Array[i])
			i++;
		else
			NotFound = false;
	}
	if( NotFound == false )
		return i;
	else
		return -1;
}

int main()
{
	int Number[] = { 67, 278, 463, 2, 4683, 812, 236, 38 };
	int Quantity = sizeof(Number) / sizeof(int);
	int NumberToSearch = 0;
	cout << "Enter the number to search: "; cin >> NumberToSearch;
	int i = LinearSearch(Number, Quantity, NumberToSearch);
	if(i == -1)
		cout << NumberToSearch << " was not found in the collection\n\n";
	else
	{
		cout << NumberToSearch << " is at the " << i+1;

		if( i == 0 )
			cout<< "st position of the collection\n\n";
		else if( i == 1 )
			cout<< "nd position of the collection\n\n";
		else if( i == 2 )
			cout<< "rd position of the collection\n\n";
		else
			cout<< "th position of the collection\n\n";
	}
	return 0;
}
```

Binary search

A **binary search** is an algorithm for locating the position of an element in a sorted array. It inspects the middle element of the sorted list: if equal to the sought value, then the position has been found; otherwise, the upper half or lower half is chosen for further searching based on whether the sought value is greater than or less than the middle element. The method reduces the number of elements needed to be checked by a factor of two each time, and finds the sought value if it exists in the list or if not determines "not present", in logarithmic time. A binary search is a dichotomic divide and conquer search algorithm.

```
#include<iostream.h>
#include<conio.h>
void binsearch(int ar[],int size,int ele)
```

```
{       int lb=0,ub=size-1,mid;             //lb=>lower bound,ub=>upper bound
     for(;lb<ub;)
       {
          mid=(lb+ub)/2;

          if(ar[mid]==ele)
            {
              cout<<"\n SEARCH SUCCESSFUL";
              break;
            }
          else
               if(ar[mid]<ele)
               ub=mid-1;
         else
              if(ar[mid]>ele)
                lb=mid+1;
      }
       if(ub<lb)
         cout<<"\n SEARCH UNSUCCESSFUL";
}

void sort(int ar[],int size)              //sorts the array in ascending array using bubble sort
{
     int temp;
     for(int i=0;i<size;i++)
         for(int j=0;j<size-i-1;j++)
          if(ar[j]>ar[j+1])
               {
                temp=ar[j];
                ar[j]=ar[j+1];
                ar[j+1]=temp;
               }
}
void display(int ar[],int size)
{
     for(int i=0;i<size;i++)
         cout<<'\n'<<ar[i];
}
void input(int ar[],int size)
{
     for(int i=0;i<size;i++)
         cin>>ar[i];
}
void main()
{
     clrscr();
     int size;
     cout<<"\n ENTER THE NUMBER OF ELEMENTS REQUIRED IN THE ARRAY :";
     cin>>size;
     int *ar=new int(size);
     cout<<"\n ENTER THE ELEMENTS OF THE ARRAY :\n";
     input(ar,size);         //takes the input from the array
     sort(ar,size);         //sorts the array in the ascending order
     int ele;
     cout<<"\n ENTER THE ELEMENT TO BE FOUND :\n";
```

```
    cin>>ele;
    getch();
}
```

Sorting

A **sorting algorithm** is an algorithm that puts elements of a list in a certain order

1. **Insertion sort** is a simple sorting algorithm, a comparison sort in which the sorted array (or list) is built one entry at a time.

```
#include <iostream>
#define ELEMENTS 6
void insertion_sort(int x[],int length)
{
  int key,i;
  for(int j=1;j<length;j++)
  {
     key=x[j];
     i=j-1;
     while(x[i]>key && i>=0)
     {
            x[i+1]=x[i];
        i--;
     }
     x[i+1]=key;
  }
}

int main()
{
  int A[ELEMENTS]={5,2,4,6,1,3};
  int x;
  cout<<"NON SORTED LIST:"<<endl;
  for(x=0;x<ELEMENTS;x++)
  {
      cout<<A[x]<<endl;
  }
  insertion_sort(A,ELEMENTS);
  cout<<endl<<"SORTED LIST"<<endl;
  for(x=0;x<ELEMENTS;x++)
  {
      cout<<A[x]<<endl;
  }
  return 0;
}
```

2. **Selection Sort**

Selection sort is a sorting algorithm, specifically an in-place comparison sort. It has $\underline{O}(n^2)$ complexity, making it inefficient on large lists, and generally performs worse than the similar insertion sort

```
void SelectionSort(int A[], int length)
{
   int i, j, min, minat;
       for(i = 0; i<(length-1); i++)
       {
```

```
                    minat = i;
                    min = A[i];

      for(j = i+1;j < length; j++) //select the min of the rest of array
          {
                     if(min > A[j])   //ascending order for descending reverse
                     {
                               minat = j;  //the position of the min element
                               min = A[j];
                     }
          }
          int temp = A[i];
          A[i] = A[minat];  //swap
          A[minat]=temp;
        }
}//end selection sort
```

3. Bubble Sort

Bubble sort is a simple sorting algorithm. It works by repeatedly stepping through the list to be sorted, comparing each pair of adjacent items and swapping them if they are in the wrong order

```
#include <iostream>
using namespace std;
int compare(int, int);
void sort(int[], const int);
int compare(int x, int y)
{
    return(x > y);
}
void sort(int table[], const int n)
{
int t;
    for(int i = 0; i < n; i++)
    {
        for(int j = 0; j < n-1; j++)
        {
            if(compare(table[j], table[j+1]))
              {
              t=table[j];
              table[j]=table[j+1];
              table[j+1]=t;

              }
        }
    }
}
int quantity;
int tab[100];

int main()
{
cout << "Input quantity: ";
cin >> quantity;
cout << "Input numbers: \n\n";
for (int i = 0; i < quantity; i++)
```

```
{
   int x = i;
   cout << "#" << ++x << ": ";
   cin >> tab[i];
}
cout << "\nBefore sorting: ";
for (int i = 0; i < quantity; i++)
{
    cout << tab[i] << " ";
}

cout << "\nAfter sorting: ";
sort(tab, quantity);
for(int i = 0; i < quantity; i++)
{
    cout << tab[i] << " ";
}
return 0;}
```

Country\Data	Map	Flag	Area (sq km)	Population
United States			9,629,091	272,639,608
Cameroon			475,440	15,456,092
Guatemala			108,890	12,335,580
Italy			301,230	56,735,130
Oman			212,460	2,446,645

Two-Dimensional Arrays

A 2-dimensional array is an array of arrays. In other words, it is an array where each member of the array is also an array. Consider the below table

Declaring and Initializing a 2-Dimensional Array

Like the above table, a 2-dimensional array is made rows and columns. To declare it, use double pair of a opening and closing square brackets. Here is an example:

```
int numberOfStudentsPerClass[12][50];
```

Based on this, when initializing a 2-dimensional array, make sure you provide a number of values that is less than or equal to the number of members.

Here is an example:

```
double distance[2][4] = {44.14, 720.52, 96.08, 468.78, 6.28, 68.04, 364.55, 6234.12};
```

Processing a 2-Dimensional Array

```
#include <iostream>
int main()
{
        double distance[][4] = {
                                { 44.14, 720.52,  96.08,  468.78 },
                                {  6.28,  68.04, 364.55, 6234.12 }
                        };

        // Scan the array from the 3rd to the 7th member
```

```
        cout << "Members of the array";
        for(int i = 0; i < 2; ++i)
                for(int j = 0; j < 4; ++j)
                        cout << "\nDistance [" << i << "][" << j << "]: " << distance[i][j];

        cout << endl;
        return 0;
}
```

STACKS

Stacks are commonly used Data Structures while writing code. It's concept is really simple which makes it even simpler to write it in code. Consider this situation. There are a pile of 5 Books on a Table. You want to add one book to the pile. What do you do? You simply add the book on the TOP of the pile. What if you want the third book from the new 6 book pile? You then lift each book one by one from the TOP until the third book reaches the top. Then you take the third book and replace all the others back into the pile by adding them from the TOP.

```
#include <iostream>
#define MAX 10        // MAXIMUM STACK CONTENT
class stack
{
  private:
    int arr[MAX];   // Contains all the Data
    int top;        //Contains location of Topmost Data pushed onto Stack
  public:
          stack()         //Constructor
          {
             top=-1;      //Sets the Top Location to -1 indicating an empty stack
          }
          void push(int a)  // Push ie. Add Value Function
          {
                  top++;        // increment to by 1
                  if(top<MAX)
                   {
                          arr[top]=a;  //If Stack is Vacant store Value in Array
                   }
                   else
                   {
                          cout<<"STACK FULL!!"<<endl;
                          top--;
                   }
          }
          int pop()               // Delete Item. Returns the deleted item
          {         if(top==-1)
                    {
                          cout<<"STACK IS EMPTY!!!"<<endl;
                          return NULL;
                    }
                    else
                    {         int data=arr[top];     //Set Topmost Value in data
                          arr[top]=NULL;       //Set Original Location to NULL
                          top--;              // Decrement top by 1
                          return data;         // Return deleted item
                    }
```

```
        }
};

int main()
{
 stack a;
 a.push(3);
 cout<<"3 is Pushed\n";
 a.push(10);
 cout<<"10 is Pushed\n";
 a.push(1);
 cout<<"1 is Pushed\n\n";
 cout<<a.pop()<<" is Popped\n";
 cout<<a.pop()<<" is Popped\n";
 cout<<a.pop()<<" is Popped\n";
 return 0;
}
```

Output:
3 is Pushed
10 is Pushed
1 is Pushed

1 is Popped
10 is Popped
3 is Popped

Clearly we can see that the last data pushed is the first one to be popped out.That's why a Stack is also known as a LIFO Data Structure which stands for "LastIn,First Out" and I guess you know why.

Let us see how we implemented the stack. We first created a variable called top that points to the top of the stack. It is initialised to -1 to indicate that the stack is empty. As Data is entered, the value in top increments itself and data is stored into an array arr. Now there's one drawback to this Data Structure. Here we state the Maximum number of elements as 10. What if we need more than 10 Data Elements? In that case we combine a Stack along with a Linked List which will be explained later.

1. Array implementation

The **array implementation** aims to create an array where the first element (usually at the zero-offset) is the bottom. That is, array[0] is the first element pushed onto the stack and the last element popped off. The program must keep track of the size, or the length of the stack. The stack itself can therefore be effectively implemented as a two-element structure in C:

```
typedef struct {
    int size;
    int items[STACKSIZE];
} STACK;
```

The push() operation is used both to initialize the stack, and to store values to it. It is responsible for inserting (copying) the value into the ps->items[] array and for incrementing the element counter (ps->size). In a responsible C implementation, it is also necessary to check whether the array is already full to prevent an overrun.

```
void push(STACK *ps, int x)
{
    if (ps->size == STACKSIZE) {
        cout<<"Error: stack overflow\n";
        abort();
    } else
        ps->items[ps->size++] = x;
```

```
}
```

The pop() operation is responsible for removing a value from the stack, and decrementing the value of ps->size. A responsible C implementation will also need to check that the array is not already empty.

```
int pop(STACK *ps)
{
    if (ps->size == 0){
        cout<<"Error: stack underflow\n";
        abort();
    } else
        return ps->items[--ps->size];
}
```

2. Linked list implementation

The **linked-list** implementation is equally simple and straightforward. In fact, a stack linked-list is much simpler than most linked-list implementations: it requires that we implement a linked-list where only the head node or element can be removed, or popped, and a node can only be inserted by becoming the new head node.

Unlike the array implementation, our structure typedef corresponds not to the entire stack structure, but to a single node:

```
typedef struct stack {
    int data;
    struct stack *next;
} STACK;
```

Such a node is identical to a typical linked-list node, at least to those that are implemented in C.

The push() operation both initializes an empty stack, and adds a new node to a non-empty one. It works by receiving a data value to push onto the stack, along with a target stack, creating a new node by allocating memory for it, and then inserting it into a linked list as the new head:

```
void push(STACK **head, int value)
{
    STACK *node =new STACK;  /* create a new node */

    if (node == NULL){
        cout<<"Error: no space available for node\n";
        abort();
    } else {                              /* initialize node */
        node->data = value;
        node->next = empty(*head) ? NULL : *head; /* insert new head if any */
        *head = node;
    }
}
```

A pop() operation removes the head from the linked list, and assigns the pointer to the head to the previous second node. It check whether the list is empty before popping from it:

```
int pop(STACK **head)
{
    if (empty(*head)) {                       /* stack is empty */
       cout<<"Error: stack underflow\n";
       abort();
    } else {                              /* pop a node */
        STACK *top = *head;
        int value = top->data;
        *head = top->next;
        Delete top;
        return value;
    }
}
```

Infix, Postfix and Prefix

Infix notation: X + Y

Operators are written in-between their operands. This is the usual way we write expressions. An expression such as A * (B + C) / D is usually taken to mean something like: "First add B and C together, then multiply the result by A, then divide by D to give the final answer." Infix notation needs extra information to make the order of evaluation of the operators clear: rules built into the language about operator precedence and associativity, and brackets ( ) to allow users to override these rules.

Postfix notation (also known as "Reverse Polish notation"): X Y +

Operators are written after their operands. The infix expression given above is equivalent to A B C + * D /. The order of evaluation of operators is always left-to-right, and brackets cannot be used to change this order. Because the "+" is to the left of the "*" in the example above, the addition must be performed before the multiplication. Operators act on values immediately to the left of them.

Prefix notation (also known as "Polish notation"): + X Y

Operators are written before their operands. The expressions given above are equivalent to / * A + B C D. As for Postfix, operators are evaluated left-to-right and brackets are superfluous. Operators act on the two nearest values on the right. I have again added (totally unnecessary) brackets to make this clear:
(/ (* A (+ B C)) D)

Examples:

Infix	Postfix	Prefix	Notes
A * B + C / D	A B * C D / +	+ * A B / C D	multiply A and B, divide C by D, add the results
A * (B + C) / D	A B C + * D /	/ * A + B C D	add B and C, multiply by A, divide by D
A * (B + C / D)	A B C D / + *	* A + B / C D	divide C by D, add B, multiply by A

Examples of Infix-to-Postfix Conversion

Infix expression: a+b*c-d/e*f

Token	operator stack → top	postfix string
	#	
a		a
+	#+	
b		ab
*	#+*	
c		abc
-	#+	abc*
	#	abc*+
	#-	
d		abc*+d
/	#-/	
e		abc*+de
*	#-	abc*+de/
	#-*	
f		abc*+de/f
	#-	abc*+de/f*
	#	abc*+de/f*-

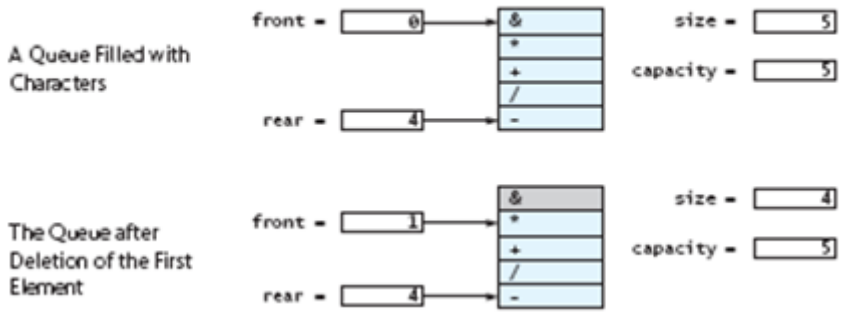
Example-1

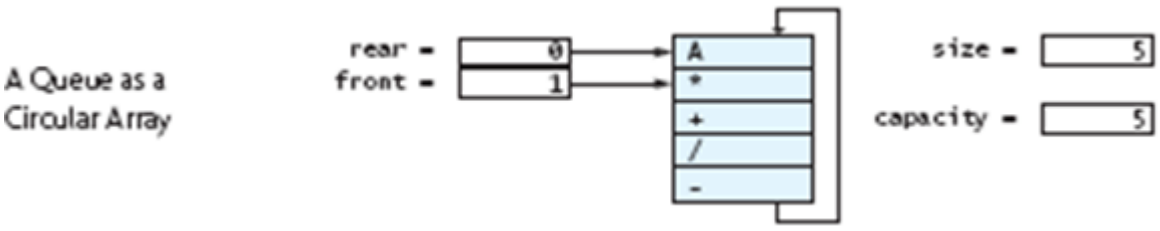
Infix expression: (a+b*c-d)/(e*f)

Token	operator stack → top	postfix string
	#	
(	#(	
a		a
+	#(+	
b		ab
*	#(+*	
c		abc
-	#(+	abc*
	#(	abc*+
	#(-	
d		abc*+d
)	#	abc*+d-
/	#/	
(	#/(	
e		abc*+d-e
*	#/(*	
f		abc*+d-ef
)	#/(	abc*+d-ef*
	#/	
	#	abc*+d-ef*/

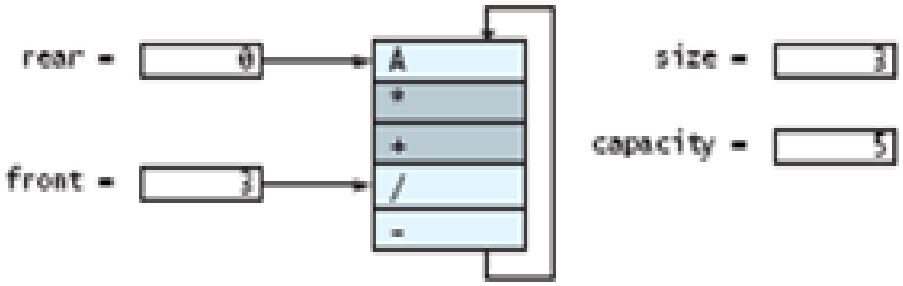
QUEUES

There's a huge crowd at your local grocery store. There are too many people trying to buy their respective items and the Shopkeeper doesnt know from where to start. Everyone wants their job done quickly and the shopkeeper needs an efficient method to solve this problem. What does he do? He introduces a Queue System based on the First Come, First Serve System. The Last Person trying to buy an item stands behind the last person at the END of the queue. The Shopkeeper however is present at the FRONT end of the queue. He gives the item to the person in FRONT of the queue and after the transaction is done, the person in FRONT of the Queue Leaves. Then the person second in queue becomes the First person in the Queue.

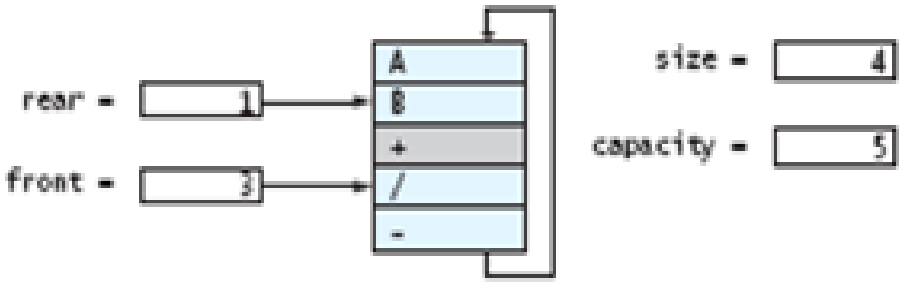




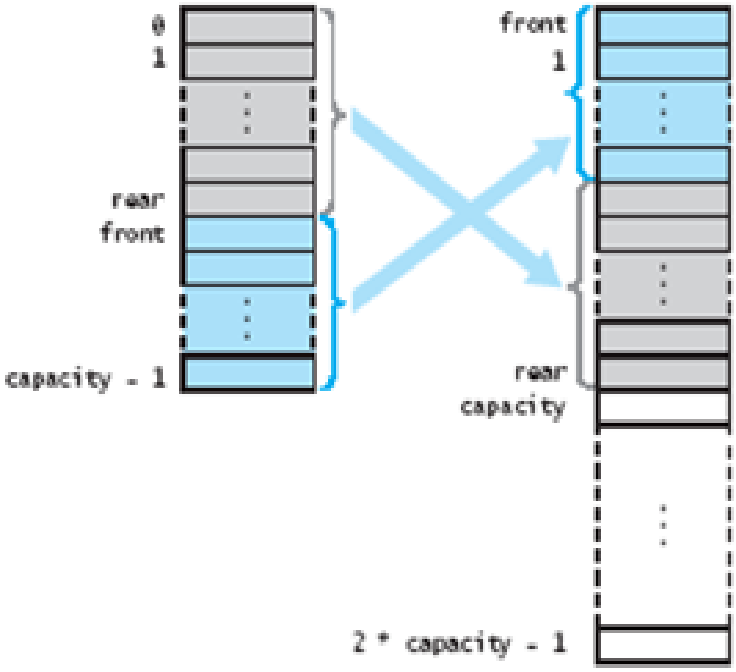
The Effect of Two Deletions . . .



and One Insertion



Reallocating a Circular Array




```
/*        QUEUE IMPLEMENTATION            */
#include <iostream>
#define MAX 5          // MAXIMUM CONTENTS IN QUEUE

class queue
{
 private:
          int t[MAX];
          int al;      // Addition End
          int dl;      // Deletion End
```

```
public:
 queue()
 {
        dl=-1;
        al=-1;
 }
 void del()
 {
   int tmp;
   if(dl==-1)
   {
          cout<<"Queue is Empty";
   }
   else
   {
              for(int j=0;j<=al;j++)
              {
                     if((j+1)<=al)
                     {
                            tmp=t[j+1];
                            t[j]=tmp;
                     }
                     else
                     {
                            al--;

                     if(al==-1)
                            dl=-1;
                     else
                            dl=0;
                     }
              }
   }
 }
void add(int item)
{
        if(dl==-1 && al==-1)
        {
              dl++;
              al++;
        }
  else
  {
              al++;
              if(al==MAX)
        {
                     cout<<"Queue is Full\n";
                     al--;
                     return;
              }
        }
        t[al]=item;

}
```

```
  void display()
  {
        if(dl!=-1)
   {
        for(int i=0;i<=al;i++)
        cout<<t[i]<<" ";
   }
   else
        cout<<"EMPTY";
  }

};

int main()
{
 queue a;
 int data[5]={32,23,45,99,24};

 cout<<"Queue before adding Elements: ";
 a.display();
 cout<<endl<<endl;

 for(int i=0;i<5;i++)
 {
   a.add(data[i]);
   cout<<"Addition Number : "<<(i+1)<<" : ";
   a.display();
   cout<<endl;
 }
 cout<<endl;
 cout<<"Queue after adding Elements: ";
 a.display();
 cout<<endl<<endl;

 for(i=0;i<5;i++)
 {
   a.del();
   cout<<"Deletion Number : "<<(i+1)<<" : ";
   a.display();
   cout<<endl;
 }
 return 0;
}
```

OUTPUT:

```
Queue before adding Elements: EMPTY

Addition Number : 1 : 32
Addition Number : 2 : 32 23
Addition Number : 3 : 32 23 45
Addition Number : 4 : 32 23 45 99
Addition Number : 5 : 32 23 45 99 24

Queue after adding Elements: 32 23 45 99 24

Deletion Number : 1 : 23 45 99 24
```

Deletion Number : 2 : 45 99 24
Deletion Number : 3 : 99 24
Deletion Number : 4 : 24
Deletion Number : 5 : EMPTY

As you can clearly see through the output of this program that addition is always done at the end of the queue while deletion is done from the front end of the queue

QUEUES WITH LINKED LIST IMPLEMENTATION

Similar to the one above, the queued linked list removes the maximum data limit as well. Here is the code:

```
#include <iostream>
struct node
{
   int data;
   node *link;
};

class lqueue
{
 private:
    node *front,*rear;
 public:
    lqueue()
    {
          front=NULL;
          rear=NULL;
    }

    void add(int n)
    {
          node *tmp;
          tmp=new node;
          if(tmp==NULL)
             cout<<"\nQUEUE FULL";

          tmp->data=n;
          tmp->link=NULL;
          if(front==NULL)
          {
                    rear=front=tmp;
                    return;
          }
          rear->link=tmp;
          rear=rear->link;
    }
    int del()
    {
          if(front==NULL)
          {
                    cout<<"\nQUEUE EMPTY";
                    return NULL;
```

```
        }
      node *tmp;
      int n;
      n=front->data;
      tmp=front;
      front=front->link;
      delete tmp;
      return n;
   }

   ~lqueue()
   {
        if(front==NULL)
          return;
        node *tmp;
        while(front!=NULL)
        {
                tmp=front;
                front=front->link;
                delete tmp;
        }
    }
};

int main()
{
        lqueue q;
        q.add(11);
        q.add(22);
        q.add(33);
        q.add(44);
        q.add(55);
        cout<<"\nItem Deleted = "<<q.del();
        cout<<"\nItem Deleted = "<<q.del();
        cout<<"\nItem Deleted = "<<q.del();
        return 0;
}
```

CIRCULAR QUEUES WITH ARRAY IMPLEMENTATION

```
#include<iostream.h>
#include<stdlib.h>
#include<stdio.h>
#include<conio.h>

// Defining class CQUEUE
class cqueue
 {
   int q[10],num,front,rear;
   public :
        cqueue();
        void insert();
        void remove();
        void menu();
        void display();
```

```
};

cqueue :: cqueue()
{
  front=rear=0;
}

void cqueue :: insert()
{
  if(((rear+1)%10)==front)
   {
     cout<<"Queue is full";
   }
  else
   {
     cout<<"Please enter a number : ";
     cin>>
     q[rear];
     rear=(rear+1)%10;
   }
}

void cqueue :: remove()
{
  if(rear==front)
   {
     cout<<"Queue is empty";
   }
  else
   {
     int num=q[front];
     cout<<"You deleted "<<num<<"";
     front=(front+1)%10;
     getch();
   }

}
void cqueue::display()
{
  int i=front;
  if(front==rear)
   {
     cout<<"Queue is empty, No elements to display !!!!!!  ";
   }
  else
   {
     cout<<"Queue's elements are :";
     cout<<"Front = ";
     while( i!=rear)
      {
          if(i==(rear-1)) cout<<"Rear = ";
          cout<<q[i]<<"";
          i=i++%10;

      } // end while.
```

```
    }// end elseif.
   getch();
 }

void cqueue :: menu()
 {
   int ch=1;
   clrscr();
   while (ch)
    {
      clrscr();
      cout<<"Enter your Choice
1 : insert
2 : remove
3 : display
0 :
exit";

      cin >>ch;
      switch (ch)
       {
            case 1 : insert();
                     break;
            case 2 : remove();
                     break;
            case 3 : display();
                     break;
            case 0 : exit(0);
       }
    }
 }
void main()
 {
  cout<<"Program to demonstrate Circular Queue";
  cqueue q1;
  q1.menu();
 }
```

UNIT 3 : DATABASES AND SQL

DATA BASE CONCEPT

- **Table:** A Table is used to store Data
- **View:** A view is the temporary table created using Original table.
- **Sequence:** Sequences are used to generate Primary key value.
- **Index:** They are used to improve queries.
- **Synonym:** They give alternative names to objects.
- **Primary Key:** The primary key of a relational table uniquely identifies each record in the table. Primary keys may consist of a single attribute or multiple attributes in combination.
- **Candidate Key:** A candidate key is a combination of attributes that can be uniquely used to identify a database record without any extraneous data. Each table may have one or more candidate keys.
- **Alternate Key:** An alternate key (or secondary key) is any candidate key which is not selected to be the primary key.
- **Foreign Key:** A foreign key is a field (or fields) that points to the primary key of another table. The purpose of the foreign key is to ensure referential integrity of the data. In other words, only values that are supposed to appear in the database are permitted.
- **Tuples:** The rows of tables (relations) are generally referred to as tuples.
- **Attribute:** The columns of tables are generally referred to as attribute.
- **Degree:** The number of attributes in a relation determines the degree of a relation.
- **Cardinality:** The number of rows in a relation is known as Cardinality.
- **Constraints:** Constraints are used to enforce rules at table level when ever row is inserted, updated/deleted from table.
- **Column Alias:** In many cases heading table may not be descriptive and hence it difficult to understand. In such case we use columns alias. It will change column heading with column alias.
- **DBA:** The DBA must be a manager, more than a technician-seeking to meet the needs of people who use the data. Since many user may share the same data resource, the DBA must be prepared to meet the need and objective.
- **DDL:** The DDL provides statements for the creation and deletion of tables and indexes.
- **DML:** The DML provides statements to enter, update, delete data and perform complex queries on these tables.
- **Select Operation:** The select operation selects tuples from a relation which satisfy a given condition. It is denoted by lowercase Greek Letter σ (sigma).
- **Project Operation:** The project operation selects columns from a relation which satisfy a given condition. It is denoted by lowercase Greek Letter π (pi). It can be thought of as picking a sub set of all available columns.
- **Union Operation:** The union (denoted as ∪) of a collection of relations is the set of all distinct tuples in the collection. It is a binary operation that needs two relations.
- **Set Difference Operation:** This is denoted by – (minus) and is a binary operation. It results in a set of tuples that are in one relation but not in another.

Tables

In relational database systems (DBS) data are represented using tables (relations). A query issued against the DBS also results in a table. A table has the following structure:
Column 1 Column 2 . . . Column n
− Tuple (or Record)
A table can have up to 254 columns which may have different or same data types and sets of values (domains), respectively. Possible domains are alphanumeric data (strings), numbers and date formats. Oracle offers the following basic data types:

• **char(n):** Fixed-length character data (string), n characters long. The maximum size for n is 255 bytes (2000 in Oracle8). Note that a string of type char is always padded on right with blanks to full length of n. (+ can be memory consuming).
Example: char(40)

• **varchar2(n):** Variable-length character string. The maximum size for n is 2000 (4000 in Oracle8). Only the bytes used for a string require storage. Example: varchar2(80)

• number(o, d): Numeric data type for integers and reals. o = overall number of digits, d = number of digits to the right of the decimal point. Maximum values: o =38, d= −84 to +127. Examples: number(8), number(5,2) Note that, e.g., number(5,2) cannot contain anything larger than 999.99 without resulting in an error. Data types derived from number are int[eger], dec[imal], smallintand real.

• **date:** Date data type for storing date and time.The default format for a date is: DD-MMM-YY. Examples: '13-OCT-94', '07-JAN-98'

• **long:** Character data up to a length of 2GB. Only one long column is allowed per table.Note: In Oracle-SQL there is no data type boolean. It can, however, be simulated by usingeither char(1) or number(1).As long as no constraint restricts the possible values of an attribute, it may have the specialvalue null (for unknown). This value is different from the number 0, and it is also different from the empty string ''.

Properties of tables are:

• the order in which tuples appear in a table is not relevant (unless a query requires an explicit sorting).

• a table has no duplicate tuples (depending on the query, however, duplicate tuples can appear in the query result). A database schema is a set of relation schemas. The extension of a database schema at database run-time is called a database instance or database, for short.

The table EMP is used to store information about employees:

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	DEPTNO
7369	SMITH	CLERK	7902	17-DEC-80	800	20
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	30

.....

Queries

In order to retrieve the information stored in the database, the SQL query language is used. In SQL a query has the following (simplified) form (components in brackets [ ] are optional):

select [distinct] <column(s)>
from <table>
[where <condition>]
[order by <column(s) [asc|desc]>]

Selecting Columns

The columns to be selected from a table are specified after the keyword select. This operation is also called projection. For example, the query select LOC, DEPTNO from DEPT; lists only the number and the location for each tuple from the relation DEPT. If all columns should be selected, the asterisk symbol "*" can be used to denote all attributes. The query

Select * from EMP;

retrieves all tuples with all columns from the table EMP. Instead of an attribute name, the select clause may also contain arithmetic expressions involving arithmetic operators etc.

select ENAME, DEPTNO, SAL - 100 from EMP;

For the different data types supported in Oracle, several operators and functions are provided:
• for numbers: abs, cos, sin, exp, log, power, mod, sqrt, +,−, _, /, . . .
• for strings: chr, concat(string1, string2), lower, upper, replace(string, search string, replacement string), translate, substr(string, m, n), length, to date, . . .
• for the date data type: add month, month between, next day, to char, . . .

select DEPTNO from EMP;

For example, the query

select ENAME, DEPTNO, HIREDATE from EMP;
from EMP
order by DEPTNO [asc], HIREDATE desc;

displays the result in an ascending order by the attribute DEPTNO. If two tuples have the same attribute value for DEPTNO, the sorting criteria is a descending order by the attribute values of HIREDATE. For the above query, we would get the following output:

ENAME DEPTNO HIREDATE

FORD 10 03-DEC-81
SMITH 20 17-DEC-80
BLAKE 30 01-MAY-81
WARD 30 22-FEB-81
ALLEN 30 20-FEB-81
.....
Selection of Tuples

If one is interested in tuples that satisfy certain conditions, the where clause is used. In a where clause simple conditions based on comparison operators can be combined using the logical connectives and, or, and not to form complex conditions. Conditions may also include pattern matching operations and even subqueries
Example: List the job title and the salary of those employees whose manager has the number 7698 or 7566 and who earn more than 1500:

select JOB, SAL
from EMP
where (MGR = 7698 or MGR = 7566) and SAL > 1500;

For all data types, the comparison operators =, != or <>,<, >,<=, => are allowed in the conditions of a where clause.
Further comparison operators are:
• Set Conditions: <column> [not] in (<list of values>)
Example: select _ from DEPT where DEPTNO in (20,30);

• Null value: <column> is [not] null,
i.e., for a tuple to be selected there must (not) exist a defined value for this column.
Example: select _ from EMP where MGR is not null;
Note: the operations = null and ! = null are not defined!

• Domain conditions: <column> [not] between <lower bound> and <upper bound>
Example: • select EMPNO, ENAME, SAL from EMP
where SAL between 1500 and 2500;
• select ENAME from EMP
where HIREDATE between ’02-APR-81’ and ’08-SEP-81’;

Some Solved Problems:
Write a query on the customers table whose output will exclude all customers with a recharge <=100, unless they are resident of Jamnagar.
SELECT * FROM customers WHERE recharge >100 OR city ='Jamnagar' ;
Write a query that selects all students except those who have not paid the fees or NULLs in the fees field.
SELECT * FROM student WHERE fees < >0 OR (fees IS NOT NULL) ;
Write a query that lists customers in descending order of purchase amount. Output the purchase amount, followed by the customer's name and number.
SELECT pur_amt, c_name, c_num FROM customers ORDER BY pur_amt DESC ;
Write a command that puts the following values, in their given order, into the salesman table:
cust-name-Ambuja, city-Ahmedabad, comm - NULL, cust-num-1001.
INSERT INTO salesman (cust-name, city, comm, cust-num)
VALUES('Ambuja','Ahmedabad',NULL,1001) ;

What is the difference between Where and Having Clause ?

The HAVING clause places the condition on group but WHERE clause places the condition on individual rows

String Operations

In order to compare an attribute with a string, it is required to surround the string by apostrophes, e.g., where LOCATION = 'DALLAS'. A powerful operator for pattern matching is the like operator. Together with this operator, two special characters are used: the percent sign % (also called wild card), and the underline , also called position marker. For example, if one is interested in all tuples of the table DEPT that contain two C in the name of the department, the condition would be where DNAME like '%C%C%'. The percent sign means that any (sub)string is allowed there, even the empty string. In contrast, the underline stands for exactly one character. Further string operations are:

• upper(<string>) takes a string and converts any letters in it to uppercase, e.g., DNAME = upper(DNAME) (The name of a department must consist only of upper case letters.)
• lower(<string>) converts any letter to lowercase,
• initcap(<string>) converts the initial letter of every word in <string> to uppercase.
• length(<string>) returns the length of the string.
• substr(<string>, n [, m]) clips out a m character piece of <string>, starting at position n.

Aggregate Functions
Aggregate functions are statistical functions such as count, min, max etc. They are used to compute a single value from a set of attribute values of a column: count Counting Rows

Example: How many tuples are stored in the relation EMP?
select count(*) from EMP;

Example: How many different job titles are stored in the relation EMP?
select count(distinct JOB) from EMP;

max Maximum value for a column
min Minimum value for a column

Example: List the minimum and maximum salary.
select min(SAL), max(SAL) from EMP;

Example: Compute the difference between the minimum and maximum salary.
select max(SAL) - min(SAL) from EMP;

sum Computes the sum of values (only applicable to the data type number)

Example: Sum of all salaries of employees working in the department 30.
select sum(SAL) from EMP where DEPTNO = 30;

avg Computes average value for a column (only applicable to the data type number)

Note: avg, min and max ignore tuples that have a null value for the specified attribute, but count considers null values.

Creating Tables
The SQL command for creating an empty table has the following form:
create table <table> (
<column 1> <data type> [not null] [unique] [<column constraint>],
. . . . . .
<column n> <data type> [not null] [unique] [<column constraint>],
[<table constraint(s)>]
);
For each column, a name and a data type must be specified and the column name must be unique within the table definition. Column definitions are separated by comma. There is no difference between names in lower case letters and names in upper case letters. In fact, the only place where upper and lower case letters matter are strings comparisons. A not null Constraint is directly specified after the data type of the column and the constraint requires defined attribute values for that column, different from null. The keyword unique specifies that no two tuples can have the same attribute value for this column.
Example: The create table statement for our EMP table has the form

```
create table EMP (
        EMPNO number(4) not null,
        ENAME varchar2(30) not null,
        JOB varchar2(10),
        MGR number(4),
        HIREDATE date,
        SAL number(7,2),
        DEPTNO number(2) );
```

Remark: Except for the columns EMPNO and ENAME null values are allowed.

Constraints

- The definition of a table may include the specification of integrity constraints.
- Basically two types of constraints are provided: column constraints are associated with a single column whereas table constraints are typically associated with more than one column.

The specification of a (simple) constraint has the following form:
[constraint <name>] primary key | unique | not null. The most important type of integrity constraints in a database are primary key constraints. A primary key constraint enables a unique identification of each tuple in a table. Based on a primary key, the database system ensures that no duplicates appear in a table. For example, for our EMP table, the specification
create table EMP (
EMPNO number(4) constraint pk emp primary key,. . .);

defines the attribute EMPNO as the primary key for the table. Each value for the attribute EMPNO thus must appear only once in the table EMP. A table, of course, may only have one primary key. Note that in contrast to a unique constraint, null values are not allowed.

Example:
We want to create a table called PROJECT to store information about projects. For each project, we want to store the number and the name of the project, the employee number of the project's manager, the budget

and the number of persons working on the project, and the start date and end date of the project. Furthermore, we have the following conditions:
- a project is identified by its project number,
- the name of a project must be unique,
- the manager and the budget must be defined.
Table definition:

```
create table PROJECT (
PNO number(3) constraint prj pk primary key,
PNAME varchar2(60) unique,
PMGR number(4) not null,
PERSONS number(5),
BUDGET number(8,2) not null,
PSTART date,
PEND date);
```

Data Modifications in SQL
After a table has been created using the create table command, tuples can be inserted into the table, or tuples can be deleted or modified.
Insertions
The most simple way to insert a tuple into a table is to use the insert statement
insert into <table> [(<column i, . . . , column j>)]
values (<value i, . . . , value j>);

Examples:

```
insert into PROJECT(PNO, PNAME, PERSONS, BUDGET, PSTART)
values(313, 'DBS', 4, 150000.42, '10-OCT-94');
or
insert into PROJECT
values(313, 'DBS', 7411, null, 150000.42, '10-OCT-94', null);
```

If there are already some data in other tables, these data can be used for insertions into a new table. For this, we write a query whose result is a set of tuples to be inserted. Such an insert statement has the form

insert into <table> [(<column i, . . . , column j>)] <query>
Example: Suppose we have defined the following table:

```
create table OLDEMP (
ENO number(4) not null,
HDATE date);
```

We now can use the table EMP to insert tuples into this new relation:

```
insert into OLDEMP (ENO, HDATE)
select EMPNO, HIREDATE from EMP
where HIREDATE < '31-DEC-60';
```

Updates
For modifying attribute values of (some) tuples in a table, we use the update statement:
update <table> set <column i> = <expression i>, . . . , <column j> = <expression j>
[where <condition>];
ypically, however, only a (small) portion of the table requires an update.
Examples:

• The employee JONES is transfered to the department 20 as a manager and his salary is increased by 1000:

```
update EMP set
JOB = 'MANAGER', DEPTNO = 20, SAL = SAL +1000
where ENAME = 'JONES';
```

• All employees working in the departments 10 and 30 get a 15% salary increase.

```
update EMP set
SAL = SAL* 1.15 where DEPTNO in (10,30);
```

In such a case we have a <query> instead of an <expression>.
Example: All salesmen working in the department 20 get the same salary as the manager who has the lowest salary among all managers.

```
update EMP set
SAL = (select min(SAL) from EMP
where JOB = 'MANAGER')
where JOB = 'SALESMAN' and DEPTNO = 20;
```

Deletions

All or selected tuples can be deleted from a table using the delete command:

```
delete from <table> [where <condition>];
```

If the where clause is omitted, all tuples are deleted from the table. An alternative command for deleting all tuples from a table is the truncate table <table> command. However, in this case, the deletions cannot be undone.
Example:

Delete all projects (tuples) that have been finished before the actual date (system date):

```
delete from PROJECT where PEND < sysdate;
```

sysdate is a function in SQL that returns the system date.

Joining Relations

- Comparisons in the where clause are used to combine rows from the tables listed in the fromclause.

Example: In the table EMP only the numbers of the departments are stored, not their name. For each salesman, we now want to retrieve the name as well as the number and the name of the department where he is working:

```
select ENAME, E.DEPTNO, DNAME
from EMP E, DEPT D
where E.DEPTNO = D.DEPTNO
and JOB = 'SALESMAN';
```

Explanation: E and D are table aliases for EMP and DEPT, respectively. The computation of the query result occurs in the following manner (without optimization):
1. Each row from the table EMP is combined with each row from the table DEPT (this operation is called Cartesian product). If EMP contains m rows and DEPT contains n rows, we thus get n * m rows.
2. From these rows those that have the same department number are selected (where E.DEPTNO = D.DEPTNO).
3. From this result finally all rows are selected for which the condition JOB = 'SALESMAN' holds.

Grouping

aggregate functions can be used to compute a single valuefor a column. Often applications require grouping rows that have certain properties and thenapplying an aggregate function on one column for each group separately. For this, SQL provides
the clause group by <group column(s)>. This clause appears after the where clauseand must refer to columns of tables listed in the from clause.

```
select <column(s)>
from <table(s)>
where <condition>
group by <group column(s)>
[having <group condition(s)>];
```

Those rows retrieved by the selected clause that have the same value(s) for <group column(s)>are grouped. Aggregations specified in the select clause are then applied to each group separately.It is important that only those columns that appear in the <group column(s)> clausecan be listed without an aggregate function in the select clause !

Example: For each department, we want to retrieve the minimum and maximum salary.

```
select DEPTNO, min(SAL), max(SAL)
from EMP
group by DEPTNO;
```

Rows from the table EMP are grouped such that all rows in a group have the same department number. The aggregate functions are then applied to each such group. We thus get the following query result:

DEPTNO	MIN(SAL)	MAX(SAL)
10	1300	5000
20	800	3000
30	950	2850

Example: Retrieve the minimum and maximum salary of clerks for each department having more than three clerks.

```
select DEPTNO, min(SAL), max(SAL)
from EMP
where JOB = 'CLERK'
group by DEPTNO
having count(*) > 3;
```

Note that it is even possible to specify a subquery in a having clause. In the above query, for example, instead of the constant 3, a subquery can be specified.

A query containing a group by clause is processed in the following way:

1. Select all rows that satisfy the condition specified in the where clause.
2. From these rows form groups according to the group by clause.
3. Discard all groups that do not satisfy the condition in the having clause.
4. Apply aggregate functions to each group.
5. Retrieve values for the columns and aggregations listed in the select clause.

UNIT 4: BOOLEAN LOGIC

Key points

Binary Decision :

- Every day we have to make logic decisions: “Should I use calculator or not?” Should I come or not?” Each of these require YES or NO, so decision which results into either YES (TRUE) or NO (FALSE) is called BINARY DECISION.
- Sentences, which can be determined to be true or false, are called logical statement and the result TRUE or False are called Truth-values.

Truth table:

- Truth table is a table, which represents all the possible values of logical variables/statements along with all the possible results of given combinations of values.

Logical Operators :

- Logical operators are derived from the Boolean algebra, which is the mathematical way of representing the concepts without much bothering about what the concepts generally means.

(a) **NOT Operator**—Operates on single variable. It gives the complement value of variable.

X	$\overline{X}$
0	1
1	0

(b) **OR Operator** -enotes Operations “logical Addition” and we use ”+” symbol

0 + 0 = 0
0 + 1 = 1
1 + 0 = 1
1 + 1 = 1

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

(c**) AND Operator** – AND Operator performs logical multiplications and symbol is (.) dot.

0.0=0
0.1=0
1.0=0
1.1=1

Truth table:

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

Evaluation of Boolean Expression Using Truth Table:

Logical variable are combined by means of logical operator (AND, OR, NOT) to form a Boolean expression.

Basic Logic Gates

A gate is simply an electronic circuit, which operates on one or more signals to produce an output signal. Gates are digital circuits because the input and output signals are either low (0) or high (1). Gates also called logic circuits.

There are three types of logic gates:

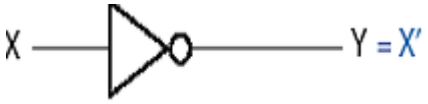
1. Inverter (NOT gate)
2. OR gate
3. AND gate

Inverter (NOT gate): Is a gate with only one input signal and one output signal, the output state is always the opposite of the input state.

- **Truth table:**

X	$\overline{X}$
Low	High
High	Low

X	$\overline{X}$
0	1
1	0



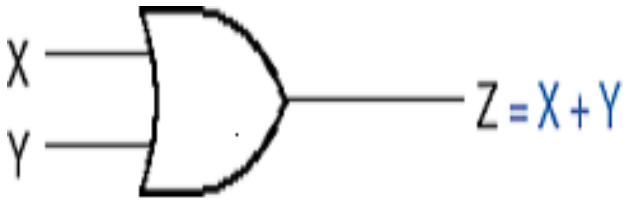
NOT gate symbol

OR gate : The OR gate has two or more input signals but only one output signal if any of the input signal is 1(high) the output signal is 1(high).

Two Input OR gate..

X	Y	F
0	0	0
0	1	1
1	0	1
1	1	1

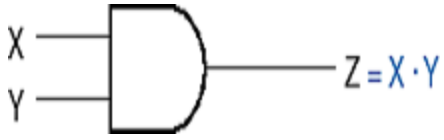
Three input OR gate



Two Input OR Gate (Circuit Diagram)

AND gate The AND gate have two or more than two input signals and produce an output signal. When all the inputs are 1(High) then the output is 1 otherwise output is 0 only.

X	Y	Z	F=X.Y.Z
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1



Circuit diagram of Two input AND gate

X	Y	Z	F=X+Y+Z
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Basic postulates of Boolean Algebra:

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

I If X != 0 then x=1 and If X!=1 then x=0

II OR relations(logical addition)

$$0 + 0 = 0$$
$$0 + 1 = 1$$
$$1 + 0 = 1$$
$$1 + 1 = 1$$

III AND relations (logical multiplication)

$$0.0=0$$
$$0.1=0$$
$$1.0=0$$
$$1.1=1$$

IV Complement Rules

$\overline{0} = 1$ AND $\overline{1} = 0$

Principal of Duality:

This states that starting with a Boolean relation another Boolean relation can be derived by. 1. Changing each OR sign (+) to a AND sign (.)

2. Changing each AND sign (.) to a OR sign (+)
3. Replacing each 0 by 1 and each 1 by 0.

Example:

0+0=0	then dual is	1.1=1
1+0=1	then dual is	0.1=0

Basic theorem of Boolean algebra

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

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

$$\overline{\overline{(X)}} = X$$

4. Complementarity Law

   (a) $X+\overline{X}=1$

   (b) $X.\overline{X}=0$

5. Commutative Law
   (a) X+Y=Y+X
   (b) X.Y=Y.X

6. Associative Law
   (a) X+(Y+Z)=(X+Y)+Z
   (b) X(YZ)=(XY)Z

7. Distributive Law
   (a) X(Y+Z)=XY_XZ
   (b) X=YZ=(X+Y)(X+Z)

8. Absorption Law
   (a) X+XY= X
   (b) X(X+Y)=X

Some other rules of Boolean algebra

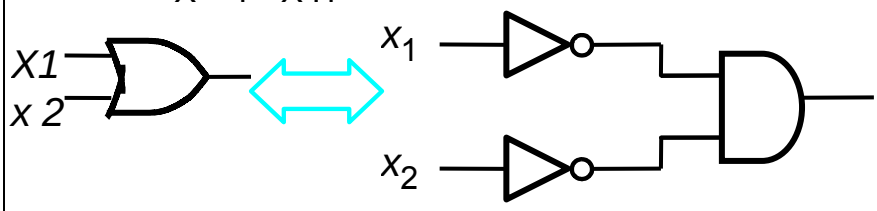
$X+X\overline{Y}=X+Y$

Demorgan's Theorem

Augustus DeMorgan had paved the way to Boolean algebra by discovering these two important theorems.

1. Demorgan's First Theorem

$\overline{X + Y} = \overline{X} . \overline{Y}$



(b) $\overline{x_1 + x_2} = \bar{x}_1 \bar{x}_2$

2. Demorgan's Second Theorem

$\overline{X.Y} = \overline{X} + \overline{Y}$

Derivation of Boolean expression

Minterms and Maxterms

- Consider two binary variables *x* and *y* combined with an AND operation.
  x'y', x'y, xy', xy

Each of these four AND terms represents one of the distinct areas in the Venn diagram and is called a *minterm* or *standard product*.

- Consider two binary variables *x* and *y* combined with an OR operation.
  x'+ y', x'+ y, x + y', x + y

Each of these four OR terms represents one of the distinct areas in the Venn diagram and is called a *maxterm* or *standard sum*.

- *n* Variables can be combined to form 2^n minterms or maxterms.

Minterms and Maxterms for Three Binary Variables						
			Minterms		Maxterms	
x	y	z	Term	Designation	Term	Designation
0	0	0	x'y'z'	m_0	x+y+z	M_0
0	0	1	x'y'z	m_1	x+y+z'	M_1
0	1	0	x'yz'	m_2	x+y'+z	M_2
0	1	1	x'yz	m_3	x+y'+z'	M_3
1	0	0	xy'z'	m_4	x'+y+z	M_4
1	0	1	xy'z	m_5	x'+y+z'	M_5
1	1	0	xyz'	m_6	x'+y'+z	M_6
1	1	1	xyz	m_7	x'+y'+z'	M_7

- A Boolean function may be represented algebraically from a given truth table by forming a minterm for each combination of the variables that produces a 1 in the function and then taking the OR of all those terms.

Functions of Three Variables				
x	***y***	***z***	**Function f_1**	**Function f_2**
0	0	0	0	0
0	0	1	1	0
0	1	0	0	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$f_1 = x'y'z + xy'z' + xyz = m_1 + m_4 + m_7$

$f_2 = x'yz + xy'z + xyz' + xyz = m_3 + m_5 + m_6 + m_7$

- The complement of a Boolean function may be read from the truth table by forming a minterm for each combination that produces a 0 in the function and then ORing those terms.

$f_1' = x'y'z' + x'yz' + x'yz + xy'z + xyz'$

Example: Express the Boolean function $F(A,B,C) = AB + C$ as a sum of minterms.

Step 1 – Each term must contain all variables
$AB = AB(C + C') = ABC + ABC'$

$C = C(A + A') = AC + A'C$

$= AC(B + B') + A'C(B + B')$

$= ABC + AB'C + A'BC + A'B'C$

Step 2 – OR all new terms, eliminating duplicates
$F(A,B,C) = A'B'C + A'BC + AB'C + ABC' + ABC$

$= m_1 + m_3 + m_5 + m_6 + m_7$
$= \Sigma(1, 3, 5, 6, 7)$

Example: Express the Boolean function $F(x,y,z) = x'y + xz$ as a product of maxterms.

Step 1 – Convert the function into OR terms using the distributive law
$F(x,y,z) = (x'y + x)(x'y + z)$

$= (x + x')(y + x)(x' + z)(y + z)$

$= (y + x)(x' + z)(y + z)$

Step 2 – Each term must contain all variables
$y + x = y + x + zz' = (x + y + z)(x + y + z')$

$x' + z = x' + z + yy' = (x' + y + z)(x' + y' + z)$

$y + z = y + z + xx' = (x + y + z)(x' + y + z)$

step 3 – AND all new terms, eliminating duplicates
$F(x,y,z) = (x + y + z)(x + y + z')(x' + y + z)(x' + y' + z)$

$= (M_0 M_1 M_4 M_6)$

$= \Pi(0, 1, 4, 6)$

Conversion Between Canonical Forms

The complement of a function expressed as the sum of minterms equals the sum of minterms missing from the original function. This is because the original function is expressed by those minterms that make the function equal to 1, whereas its complement is a 1 for those minterms that the function is 0.

Example : $F(A,B,C) = \Sigma(0, 2, 4, 6, 7)$

$F'(A,B,C) = \Sigma(1, 3, 5) = m_1 + m_3 + m_5$

Take the complement of *F'* by DeMorgan's theorem to obtain *F* in a different form:

$F(A,B,C) = (m_1 + m_3 + m_5)' = (m_1' \cdot m_3' \cdot m_5') = M_1M_3M_5 = \Pi(1, 3, 5)$

- To convert from one canonical form to the other, interchange the symbols Σ and Π, and list those numbers missing from the original form.

Standard Forms

- The two canonical forms of Boolean algebra are basic forms that one obtains from reading a function from the truth table. By definition, each minterm or maxterm must contain all variables in either complemented or uncomplemented form.
- Another way to express Boolean functions is in *standard form*. In this configuration, the terms that form the function may contain one, two, or any number of literals.
- There are two types of standard forms: the *sum of products* and the *product of sums*.
- The sum of products is a Boolean function containing AND terms, called *product terms*, of one or more literals each. The sum denotes the ORing of these terms.
  Example: $f_1 = y' + xy + x'yz'$
- The product of sums is a Boolean expression containing OR terms, called *sum terms*. Each term may have one or more literals. The product denotes the ANDing of these terms.
  Example: $f_2 = x(y' + z)(x' + y + z' + w)$
- A Boolean function may also be expressed in nonstandard form.
  Example: $f_3 = (AB + CD)(A'B' + C'D')$

Minimization of Boolean expression

A minimization Boolean expression means less numbers of gates, which means simplified circuitry.

There are two methods.

1. Algebraic Method :

This method makes use of Boolean postulates, rules and theorems to simplify the expression.

Example..1. Simplify AB`CD` + AB`CD +ABCD` +ABCD

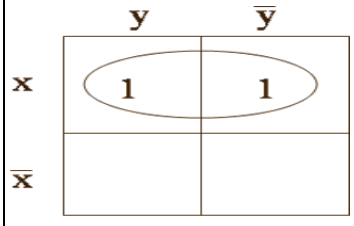
Solution-- AB`CD` + AB`CD +ABCD` +ABCD

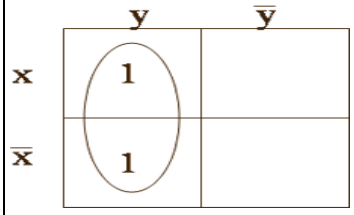
=AB`C(D`+D) +ABC(D`+D)

=AB`C.1 + ABC.1 (D`+D=1)

=AC(B`+B)

=AC.1 =AC





Example..2.. Reduce.. X`Y`Z`+X`YZ`+XY`Z`+XYZ`

Solution… X`Y`Z`+X`YZ`+XY`Z`+XYZ

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

=X`(Z`(Y`+Y)) +X(Z`(Y`+Y))

=X`(Z`.1) +X(Z`.1) (Y`+Y=1)

=X`Z`+XZ`

=Z`(X`+X)

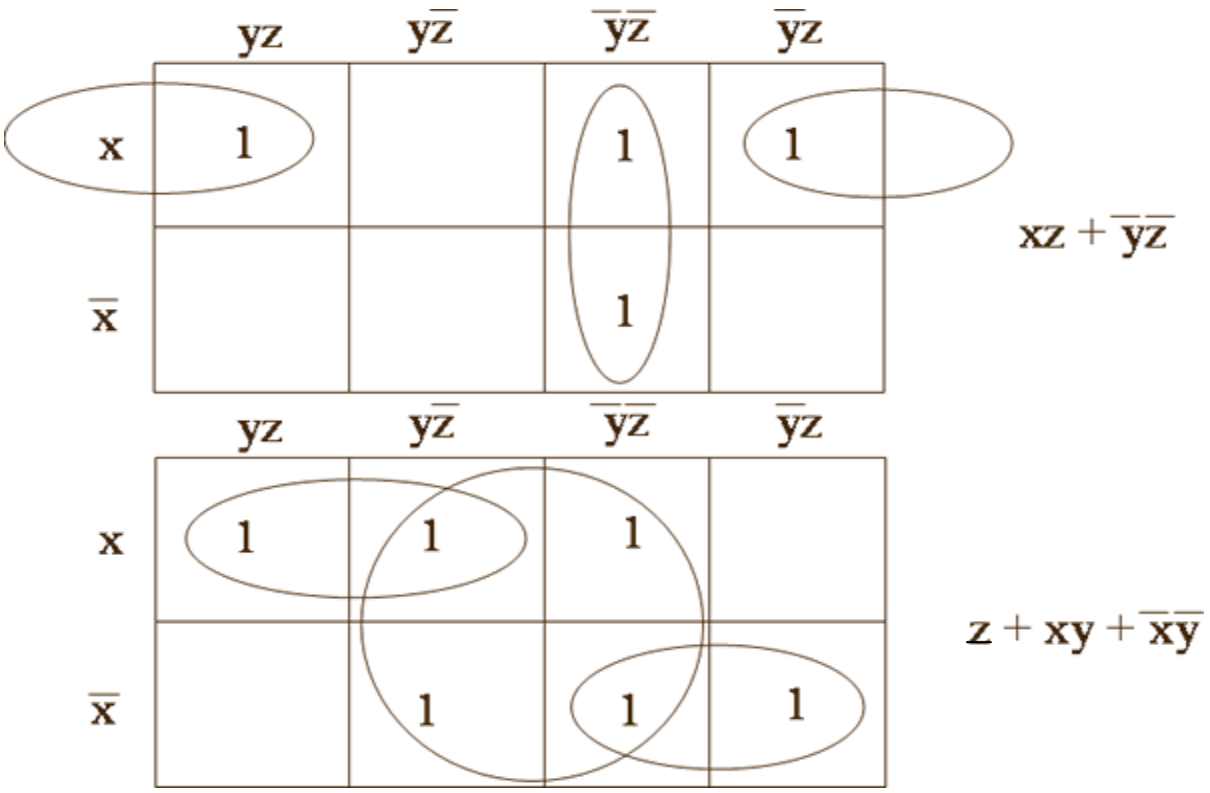
=Z`.1

=Z`

2. Using Karnaugh Map :

A Karnaugh map is graphical display of the fundamental products in a truth table.
For example:

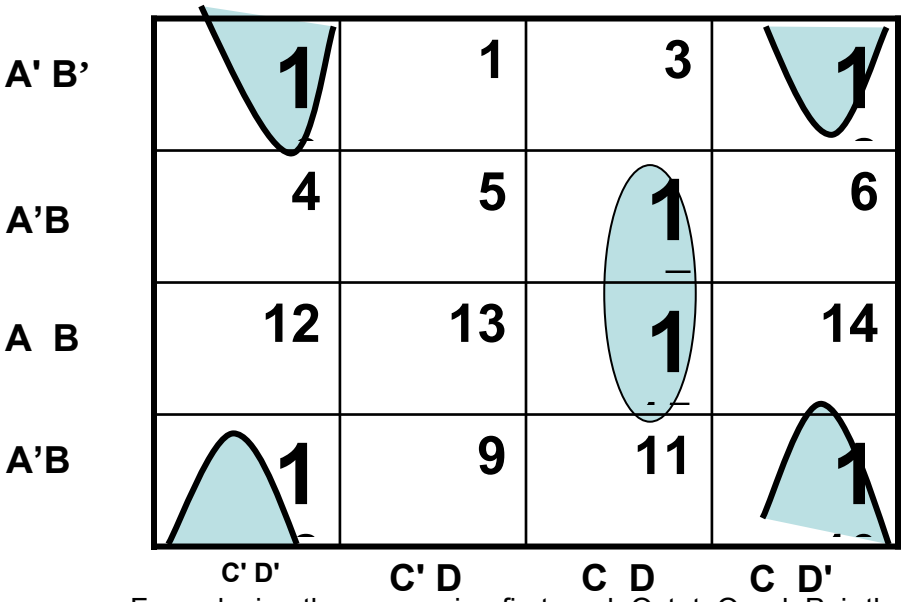
- Put a 1 in the box for any minterm that appears in the SOP expansion.
- Basic idea is to cover the **largest adjacent** blocks you can whose side length is some power of 2.
- Blocks can "wrap around" the edges.
- For example, the first K-map here represents $xy + x\overline{y} = x(y + \overline{y}) = x$. (It 'neutralizes' the *y* variable)
- The second K-map, similarly, shows $xy + \overline{x}y = (x + \overline{x})y = y$. (It 'neutralizes' the *x* variable)



- Remember, group together adjacent cells of 1s, to form largest possible rectangles of sizes that are powers of 2.
- Notice that you can overlap the blocks if necessary.

<u>Sum Of Products Reduction using K- Map</u>

- In SOP reduction each square represent a minterm.
- Suppose the given function is f(A,B,C,D) = ∑ (0,2,7,8,10,15)
- Enter the 1 in the corresponding position for each minterm of the given function.
- Now the K-map will be as follows



- For reducing the expression first mark Octet, Quad, Pair then single.
- Pair: Two adjacent 1's makes a pair.
- Quad: Four adjacent 1's makes a quad.
- Octet: Eight adjacent 1's makes an Octet.
- Pair removes one variable.
- Quad removes two variables.
- Octet removes three variables.

Reduction of Expression : When moving vertically or horizontally in pair or a quad or an octet it can be observed that only one variable gets changed that can be eliminated directly in the expression.
For Example
In the above Ex

Step 1 : In K Map while moving from m_7 to m_{15} the variable A is changing its state Hence it can be removed directly, the solution becomes **B.CD = BCD.** This can be continued for all the pairs, Quads, and Octets.

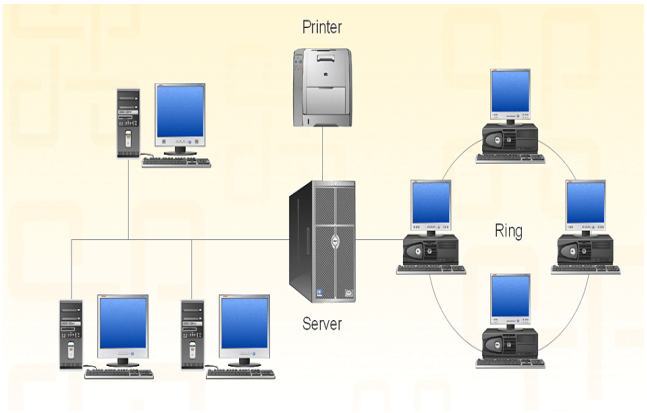
Step 2 : In K map while moving from m_0 to m_8 and m_2 to m_{10} the variable A is changing its state. Hence **B'** can be taken Similarly while moving from m_0 to m_2 and m_8 to m_{10} the variable C is changing its state. Hence **D'** can be taken , the solution becomes **B'.D'**
The solution for above expression using K map is BCD + B'D'.

UNIT 5 : COMMUNICATION AND NETWORK CONCEPTS

Points to remember

Network

- The collection of interconnected computers is called a computer network.
- Two computers are said to be interconnected if they are capable of sharing and exchanging information.



needeeds

- Resource Sharing
- Reliability
- Cost Factor
- Communication Medium

Resource Sharing means to make all programs, data and peripherals available to anyone on the network irrespective of the physical location of the resources and the user.

Reliability means to keep the copy of a file on two or more different machines, so if one of them is unavailable (due to some hardware crash or any other) them its other copy can be used.

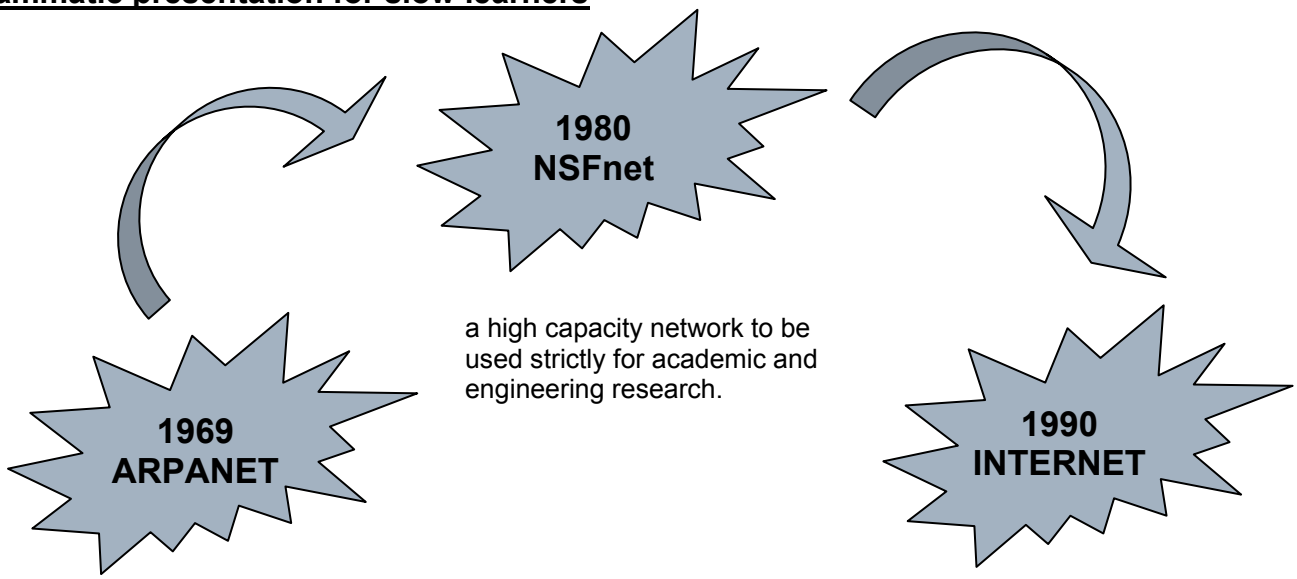
Cost factor means it greatly reduces the cost since the resources can be shared

Communication Medium means one can send messages and whatever the changes at one end are done can be immediately noticed at another.

Evolution Of Networking

1969 - First network came into existence
ARPANET (ADVANCED RESEARCH PROJECT AGENCY NETWORK)
MID 80'S - **NSFNET** (NATIONAL SCIENCE FOUNDATION NETWORK)

Diagrammatic presentation for slow learners



- **Internet** is the network of networks.

SWITCHING TECHNIQUES

Switching techniques are used for transmitting data across networks.
Different types are :

- Circuit Switching
- Message Switching
- Packet Switching

Circuit Switching

- *Circuit switching* is the transmission technology that has been used since the first communication networks in the nineteenth century.
- First the complete physical connection between two computers is established and then the data are transmitted from the source computer to the destination.
- When a call is placed the switching equipment within the system seeks out a physical copper path all the way from the sender to the receiver.
- It is must to setup an end-to-end connection between computers before any data can be sent.
- The circuit is *terminated* when the connection is closed.
- In circuit switching, resources remain allocated during the full length of a communication, after a circuit is established and until the circuit is terminated and the allocated resources are freed.

Message Switching

- In this the source computer sends data or the message to the switching circuit which stores the data in its buffer.
- Then using any free link to the switching circuit the data is send to the switching circuit.
- Entire message is sent to the destination. It reaches through different intermediate nodes following the "store and forward" approach.
- No dedicated connection is required.

Packet Switching

- Conceived in the 1960's, *packet switching* is a more recent technology than circuit switching.
- Packet switching introduces the idea of cutting data i.e. at the source entire message is broken in smaller pieces called packets which are transmitted over a network without any resource being allocated.
- Then each packet is transmitted and each packet may follow any rout available and at destination packets may reach in random order.
- If no data is available at the sender at some point during a communication, then no packet is transmitted over the network and no resources are wasted.
- At the destination when all packets are received they are merged to form the original message.
- In packet switching all the packets of fixed size are stored in main memory.

DATA COMMUNICATION TERMINOLOGIES

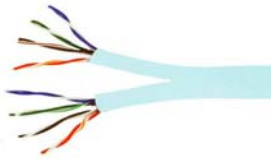
Data channel	• The information / data carry from one end to another in the network by channel.
Baud & bits per second (bps)	• It's used to measurement for the information carry of a communication channel. • Measurement Units :- • bit • 1 Byte= 8 bits • 1 KBPS (Kilo Byte Per Second)= 1024 Bytes , • 1 Kbps (kilobits Per Second) = 1024 bits, • 1 Mbps (Mega bits Per Second)=1024 Kbps
Bandwidth	• It is amount of information transmitted or receives per unit time. • It is measuring in Kbps/Mbps etc unit.

Transmission Media

- data is transmitted over copper wires, fiber optic cable, radio and microwaves. the term 'media' is used to generically refer to the physical connectors, wires or devices used to plug things together.
- **Basic communications media types**
- copper
  - unshielded twisted pair (utp)
  - shielded twisted pair (stp)
  - coaxial cable (thinnet, thicknet)
- fiber optic
  - single-mode
  - multi-mode
- infrared
- radio & microwave

Twisted Pair Cable

- These cables consist of two insulated copper wires twisted around each other in a double helix.
- Twisting of wires reduces crosstalk which is bleeding of a signal from one wire to another.



Types:

- Unshielded Twisted Pair (UTP)
- Shielded Twisted Pair (STP

STP offers greater protection from interference and crosstalk due to shielding.
But it is heavier and costlier than UTP.

USE:

1. In local telephone communication
2. For digital data transmission over short distances upto 1 km

Advantages:

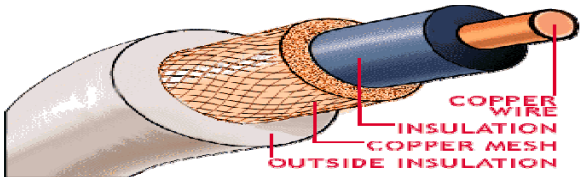
- Easy to install and maintain
- Simple
- Inexpensive
- Low weight
- Suitable for small (Local) Networks

Disadvantages:

- Not suitable for long distance due to high attenuation.
- Low bandwidth support.
- Low Speed

Coaxial cable

- Coaxial cable consists of a solid copper wire core surrounded by a plastic cladding shielded in a wire mesh.
- Shield prevents the noise by redirecting it to ground**.**



Types:

Coaxial cable comes in two sizes which are called *thinnet* and *thicknet.*

- Thicknet : segment length upto 500 m
- Thinnet : segment length upto 185 m

USE:

In TV channel communication

Advantages:

- Better than twisted wire cable.
- Popular for TV networks.
- Offers higher bandwidth & Speed

- **Disadvantages:**

- Expensive than twisted wires.
- Not compatible with twisted wire cable.

Optical Fibres

- Thin strands of glass or glass like material designed to carry light from one source to another.
- Source converts (Modulates) the data signal into light using LED (Light Emitting Diodes) or LASER diodes and send it over the Optical fiber.

It consists of three parts:

1. The core: glass or plastic through which the light travels.
2. The cladding : covers the core and reflects light back to the core
3. Protective coating : protects the fiber

Advantages

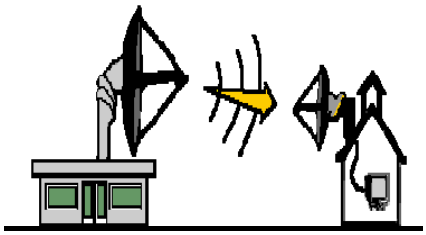
- Not affected by any kind of noise.
- High transmission capacity
- Speed of Light
- Suitable for broadband communication

Disadvantages

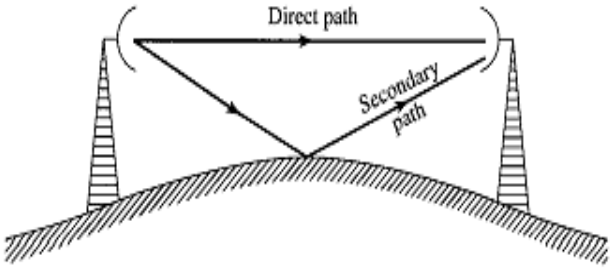
- Installation requires care.
- Connecting two Optical fibers is difficult.
- Optical fibers are more difficult to solder
- Most expensive

Microwaves

Microwaves are transmitted from the transmitters placed at very high towers to the receivers at a long distance.



Microwaves are transmitted in line of sight fashion, and also propagated through the



surfaces.

Advantages

- Maintenance easy than cables.
- Suitable when cable can not be used.
-

Disadvantages

- Repeaters are required for long distance communication.
- Less Bandwidth available.

Satellite

Geostationary satellites are placed around 36000 KM away from the earth's surface. In satellite communication transmitting station transmits the signals to the satellite. (It is called up-linking). After receiving the signals (microwaves) it amplifies them and transmit back to earth in whole visibility area. Receiving stations at different places can receive these signals. (It is called down-linking).

Advantage

- Area coverage is too large

Disadvantage

- High investment

Network devices

Modem

- A modem is a computer peripheral that allows you to connect and communicate with other computers via telephone lines.
- Modem means Modulation/ Demodulation.
- Modulation: A modem changes the digital data from your computer into analog data, a format that can be carried by telephone lines.
- Demodulation: The modem receiving the call then changes the analog signal back into digital data that the computer can digest.
- The shift of digital data into analog data and back again, allows two computers to speak with one another.

External Modem

Internal Modem

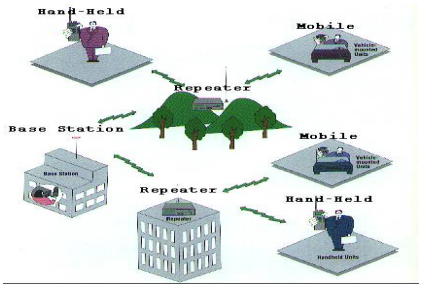
RJ- 45 Connector

RJ-45 is short for Registered Jack-45. It is an eight wire connector which is commonly used to connect computers on the local area networks i.e., LAN.

Network Interface Cards (Ethernet Card)

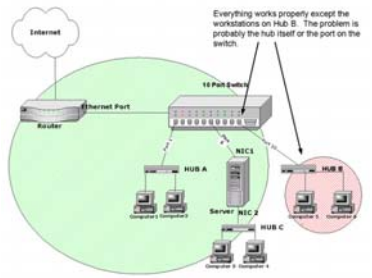
- A network card, network adapter or NIC (network interface card) is a piece of computer hardware designed to allow computers to communicate over a **computer network**. It provides physical access to a networking medium and often provides a low-level addressing system through the use of MAC addresses. It allows users to connect to each other either by using cables or wirelessly.

Repeaters



A repeater is an electronic device that receives a signal and retransmits it at a higher level or higher power, or onto the other side of an obstruction, so that the signal can cover longer distances without degradation. In most twisted pair Ethernet configurations, repeaters are required for cable runs longer than 100 meters.

Hubs



A hub contains multiple ports. When a packet arrives at one port, it is copied to all the ports of the hub. When the packets are copied, the destination address in the frame does not change to a broadcast address. It does this in a rudimentary way, it simply copies the data to all of the Nodes connected to the hub.

BridgesA network bridge connects multiple network segments at the data link layer (layer 2) of the OSI model. Bridges do not promiscuously copy traffic to all ports, as hubs do, but learn which MAC addresses are reachable through specific ports. Once the bridge associates a port and an address, it will send traffic for that address only to that port. Bridges do send broadcasts to all ports except the one on which the broadcast was received.

- Bridges learn the association of ports and addresses by examining the source address of frames that it sees on various ports. Once a frame arrives through a port, its source address is stored and the bridge assumes that MAC address is associated with that port. The first time that a previously unknown destination address is seen, the bridge will forward the frame to all ports other than the one on which the frame arrived.
- Bridges come in three basic types:
- Local bridges: Directly connect local area networks (LANs)
- Wireless bridges: Can be used to join LANs or connect remote stations to LANs.

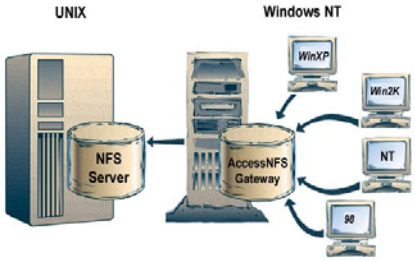
Switches

Switch is a device that performs switching. Specifically, it forwards and filters OSI layer 2 datagrams (chunk of data communication) between ports (connected cables) based on the Mac-Addresses in the packets. This is distinct from a hub in that it only forwards the datagrams to the ports involved in the communications rather than all ports connected. Strictly speaking, a switch is not capable of routing traffic based on IP address (layer 3) which is necessary for communicating between network segments or within a large or complex LAN. Some switches are capable of routing based on IP addresses but are still called switches as a marketing term. A switch normally has numerous ports with the intention that most or all of the network be connected directly to a switch, or another switch that is in turn connected to a switch.

Routers

- Routers are networking devices that forward data packets between networks using headers and forwarding tables to determine the best path to forward the packets. Routers work at the network layer of the TCP/IP model or layer 3 of the OSI model. Routers also provide interconnectivity between like and unlike media (RFC 1812).
- A router is connected to at least two networks, commonly two LANs or WANs or a LAN and its ISP's network.

GATEWAY

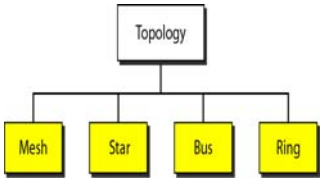


A Gateway is a network device that connects dissimilar networks. It established an intelligent connection between a local area network and external networks with completely different structures.

Network topologies and types

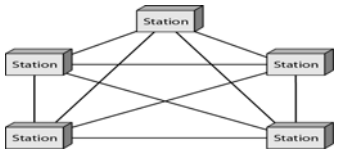
Network topology

- Computer networks may be classified according to the network topology upon which the network is based, such as Bus network, Star network, Ring network, Mesh network, Star-bus network, Tree or Hierarchical topology network, etc.
- Network Topology signifies the way in which intelligent devices in the network see their logical relations to one another. The use of the term "logical" here is significant. That is, network topology is independent of the "physical" layout of the network. Even if networked computers are physically placed in a linear arrangement, if they are connected via a hub, the network has a Star topology, rather than a Bus Topology. In this regard the visual and operational characteristics of a network are distinct; the logical network topology is not necessarily the same as the physical layout



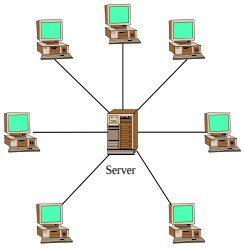
Mesh Topology

- The value of fully meshed networks is proportional to the exponent of the number of subscribers, assuming that communicating groups of any two endpoints, up to and including all the end points.

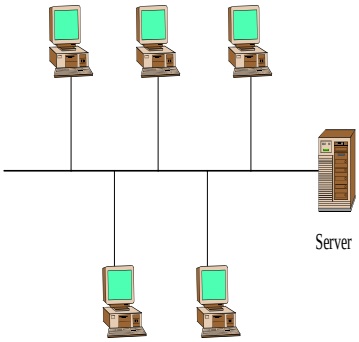


Star Topology

The type of network topology in which each of the nodes of the network is connected to a central node with a point-to-point link in a 'hub' and 'spoke' fashion, the central node being the 'hub' and the nodes that are attached to the central node being the 'spokes' (e.g., a collection of point-to-point links from the peripheral nodes that converge at a central node) – all data that is transmitted between nodes in the network is transmitted to this central node, which is usually some type of device that then retransmits the data to some or all of the other nodes in the network, although the central node may also be a simple common connection point (such as a 'punch-down' block) without any active device to repeat the signals.

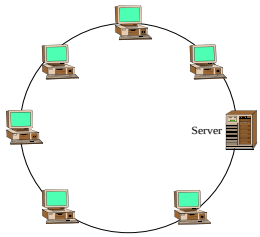


Bus Topology



- The type of network topology in which all of the nodes of the network are connected to a common transmission medium which has exactly two endpoints (this is the 'bus', which is also commonly referred to as the backbone, or trunk) – all data that is transmitted between nodes in the network is transmitted over this common transmission medium and is able to be received by all nodes in the network virtually simultaneously (disregarding propagation delays).

Ring Topology



- The type of network topology in which each of the nodes of the network is connected to two other nodes in the network and with the first and last nodes being connected to each other, forming a ring – all data that is transmitted between nodes in the network travels from one node to the next node in a circular manner and the data generally flows in a single direction only.

Compter Networks

- A communications network is two or more computers connected to share data and resources are “networked.” The simple idea behind computer networking is to allow users to access more information and give them access to devices not directly attached to their “local” system, such as printers or storage devices

Local Area Network (LAN)

A network covering a small geographic area, like a home, office, or building. Current LANs are most likely to be based on Ethernet technology. For example, a library will have a wired or a communications network is two or more computers connected to share data and resources are “networked.” The simple

idea behind computer networking is to allow users to access more information and give them access to devices not directly attached to their “local” system, such as **printers or storage devices**

Metropolitan Area Network (MAN)

- A Metropolitan Area Network is a network that connects two or more Local Area Networks or Campus Area Networks together but does not extend beyond the boundaries of the immediate town, city, or metropolitan area. Multiple routers, switches & hubs are connected to create a MAN.

Wide Area Network (WAN)

- WAN is a data communications network that covers a relatively broad geographic area (i.e. one city to another and one country to another country) and that often uses transmission facilities provided by common carriers, such as telephone companies. WAN technologies generally function at the lower three layers of the OSI reference model: the physical layer, the data link layer, and the network layer.

Network protpcol

Protocols

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

Types of protocols are:

1. HTTP
2. FTP
3. TCP/IP
4. SLIP/PPP

- **Hypertext Transfer Protocol** (**HTTP**) is a communications protocol for the transfer of information on the intranet and the World Wide Web. Its original purpose was to provide a way to publish and retrieve hypertext pages over the Internet.
- HTTP is a request/response standard between a client and a server. A client is the end-user; the server is the web site.

- **FTP (File Transfer Protocol)** is the simplest and most secure way to exchange files over the Internet. The objectives of FTP are:
- To promote sharing of files (computer programs and/or data).
- To encourage indirect or implicit use of remote computers.
- To shield a user from variations in file storage systems among different hosts.

- To transfer data reliably, and efficiently.

- **TCP/IP (Transmission Control Protocol / Internet Protocol)**

TCP - is responsible for verifying the correct delivery of data from client to server. Data can be lost in the intermediate network. TCP adds support to detect errors or lost data and to trigger retransmission until the data is correctly and completely received.

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

TCP and IP were developed by a Department of Defense (DOD) research project to connect a number different networks designed by different vendors into a network of networks.

Several computers in a small department can use TCP/IP (along with other protocols) on a single LAN. The IP component provides routing from the department to the enterprise network, then to regional networks, and finally to the global Internet.

- **SLIP/PPP (Serial Line Internet Protocol / Point to Point Protocol)**

SLIP/PPP provides the ability to transport TCP/IP traffic ever serial line between two computers. The home user's computer has a communications link to the internet. The home user's computer has the networking software that can speak TCP/IP with other computers on the Internet. The home user's computer has an identifying address (IP address) at which it can be contacted by other computers on Internet. E.g. dial up connection.

Telnet-
It is an older internet utility that lets us log on to remote computer system. It also facilitates for terminal emulation purpose. Terminal emulation means using a pc like a mainframe computer through networking.
(i) Run telnet client- Type telnet in run dialog box.
(ii) Connect to telnet site -specify the host name, port and terminal type.
(iii) Start browsing- surf the shown site with provided instruction.
(iv) Finally disconnect-press Alt+F4.

Wireless/Mobile Computing

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

1. **GSM(Global System for Mobile communication):** it is leading digital cellular system. In covered areas, cell phone users can buy one phone that will work any where the standard is supported. It uses narrowband TDMA, which allows eight simultaneous calls on the same radio frequency.

2. **CDMA(Code Division Multiple Access)**: it is a digital cellular technology that uses spread-spectrum techniques. CDMA does not assign a specific frequency to each user. Instead ,every channel uses the full available spectrum.

3. **WLL(Wireless in Local Loop) :** WLL is a system that connects subscribers to the public switched telephone network using radio signals as a substitute for other connecting media.

4. **Email(Electronic Mail):** Email is sending and receiving messages by computer.

5. **Chat:** Online textual talk in real time , is called Chatting.

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

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

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

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

NETWORK SECURITY CONCEPTS

Protection methods

1 Authorization - Authorization confirms the service requestor's credentials. It determines if the service requestor is entitled to perform that operation.
2 Authentication - Each entity involved in using a web service the requestor, the provider and the broker(if there is one) - is what it actually claims to be.
3 Encryption – conversion of the form of data from one form to another form.
4 Biometric System - involves unique aspect of a person's body such as Finger-prints, retinal patterns etc to establish his/her Identity.
5 Firewall - A system designed to prevent unauthorized access to or from a private network is called firewall. it can be implemented in both hardware and software or combination or both.

There are several types of firewall techniques-

* **Packet filter-** accepts or rejects of packets based on user defined rules.
* **Application gateway-** security mechanism to specific application like FTP and Telnet servers.
* **Circuit level gateway -** applies security mechanism when a connection is established.
* **Proxy Server -** Intercepts all messages entering and leaving the network.

Cookies - 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. Cookies have few parameters name, value, expiration date

Hackers and crackers -
Hackers are more interested in gaining knowledge about computer systems and possibly using this knowledge for playful pranks.
Crackers are the malicious programmers who break into secure systems.

Cyber Law -
It is a generic term, which refers to all the legal and regulatory aspects of internet and the World Wide Web.

WEB SERVERS

WWW (WORLD WIDE WEB)
It is a small part of Internet. It is a kind of Application of internet.It is a set of protocols that allows us to access any document on the Net through a naming system based on URLS. Internet was mainly used for obtaining textual information. But post-WWW the internet popularity grew tremendously because of

graphic intensive nature of www.

Attributes of WWW
(i) **User friendly**- www resources can be easily used with the help of browser.
(ii) **Multimedia documents**-A web page may have graphic, audio, video, and animation etc at a time.
(iii) **Hypertext and hyperlinks**-the dynamic links which can move towards another web page is hyperlink.
(iv) **Interactive** -www with its pages support and enable interactivity between users and servers.
(v) **frame**-display of more than one section on single web page.

Web server- It is a WWW server that responds to the requests made by web browers.
e.g. : Apache, IIS, PWS(Personal web server for Windows 98).

Web browser- It is a WWW client that navigates through the World Wide Web and displays web pages.
E.g.: FireFox Navigator, Internet Explorer etc.
Web sites- A location on a net server where different web pages are linked together by dynamic links is called a web site. Each web site has a unique address called URL.

Web page - A document that can be viewed in a web browser and residing on a web site is a web page.
Home page- a web page that is the starting page and acts as an indexed page is home page.
Web portal - that facilitates various type of the functionality as web site. for e.g. www.yahoo.com,www.rediff.com

Domain name- An internet address which is a character based is called a Domain name. Some most common domains are com, edu, gov, mil, net, org, and co.Some domain names are location based also. For e.g. au for Australia, a for Canada, in for India etc.

URL- A URL (uniform resource locator) that specifies the distinct address for each resource on the internet.e.g.http://encycle.msn.com/getinfo/stypes.asp

Web hosting - means hosting web server application on a computer system through which electronic content on the internet is readily available to any web browser client.

HTML -
It stands for Hyper Text Markup Language that facilitates to write web document that can be interpreted by any web browser. It provide certain tags that are interpreted by the browser how to display and act with the text, graphics etc. tags are specified in <>.
For e.g.
<body bgcolor=green> it is opening tag
</body> it is closing tag.
body is the tag with bgcolor attributes.

XML (eXtensible Markup Language)
XML is a markup language for documents containing structured information. Structured information contains both content (words, pictures etc.) and some indication of what role content plays.

DHTML- It stands for Dynamic Hyper Text Markup Language. DHTML refers to Web content that changes each time it is viewed. For example, the same URL could result in a different page depending on any number of parameters, such as:
*geographic location
*time of the day
*previous pages viewed by the user
*profile of the reader

WEB SCRIPTING – The process of creating and embedding scripts in a web page is known as web-scripting.

SCRIPT: A Script is a list of commands embedded in a web page. Scripts are interpreted and executed by a certain program or scripting –engine.

Types of Scripts:

1. **Client Side Script:** Client side scripting enables interaction within a web page.
Some popular client-side scripting languages are VBScript, JavaScript, PHP(Hyper Text Preprocessor).
2. Server-Side Scripts: Server-side scripting enables the completion or carrying out a task at the server-end and then sending the result to the client –end.
Some popula server-side Scripting Languages are PHP, Perl, ASP(Active Server Pages), JSP(Java Server Pages) etc.

OPEN SOURCE TERMINOLOGIES

TERMINOLOGY & DEFINITIONS:

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

OPEN SOURCE / FREE SOFTWARE

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

- Apache Web server is maintained by open community of developers of Apache software foundation.

- **MYSQL :** MYSQL is one of the most popular open source database system. Features of MYSQl :
  - Multithreading
  - Multi –User
  - SQl Relational Database Server
  - Works in many different platform
- **PostgreSQL :** Postgres SQL is a free software object relational database server . PostgresSQL can be downloaded from www.postgressql.org.
- **Pango :** Pango project is to provide an open source framework for the layout and rendering of internationalized text into GTK + GNOME environment.Pango using Unicode for all of its encoding ,and will eventually support output in all the worlds major languages.
- **OpenOffice :** OpenOffice is an office applications suite. It is intended to compatible and directly complete with Microsoft office.
  OOo Version 1.1 includes:
  - Writer (word processor)
  - Calc(spreadsheet)
  - Draw(graphics program)
  - etc
- **Tomcat :** Tomcat functions as a servlet container. Tomcat implements the servlet and the JavaServer Pages .Tomcat comes with the jasper compiler that complies JSPs into servlets.
- **PHP(Hypertext Preprocessor) :** PHP is a widely used open source programming language for server side application and developing web content.
- **Python: Python** is an interactive programming language originally as scripting language for Amoeba OS capable of making system calls.

Important Board Questions MARKS

1. Explain function of hub and router. 1

Ans:

1**. Hub:** A hub contains multiple ports. When a packet arrives at one port, it is copied to all the ports of the hub. When the packets are copied, the destination address in the frame does not change to a broadcast address. It does this in a rudimentary way, it simply copies the data to all of the Nodes connected to the hub.

2. **Router :** routers are networking devices that forward data packets between networks using headers and forwarding tables to determine the best path to forward the packets

2. Expand the following terms 2
   (i) URL (ii) ISP (iii) DHTML (iv) CDMA:

Ans; (i) URL: Unified Resource Locator
(ii) ISP: Internet Service Provider.
(iii) DHTML: Dynamic Hyper Text Markup Language

3. Differentiate between message switching and packet switching 1

Ans: Message Switching – In this form of switching no physical copper path is established in advance between sender and receiver. Instead when the sender has a block of data to be sent, it is stored in first switching office, then forwarded later. Packet Switching – With message switching there is no limit on block size, in contrast packet switching places a tight upper limit on block size.

4. Write two applications of Cyber Law. 2

Ans: Two applications of cyber law are Digital Transaction and Activities on Internet.

5. Explain GSM. 1

Ans: Global system for mobile, communications is a technology that uses narrowband TDMA, which allows eight simultaneous calls on the same radio frequency. TDMA is short for Time Division Multiple Access. TDMA technology uses time division multiplexing and divides a radio frequency into time slots and then allocates these slots to multiple calls thereby supporting multiple, simultaneous data channels.

6. Write difference between coaxial and optical cable. 1

Ans: Coaxial cable consists of a solid wire core surrounded by one or more foil or wire shield , each separated by some kind of plastic insulator. Optical fibers consists of thin strands of glass or glass like material which are so constructed that they carry light from a source at one end of the fiber to a detector at the other end.

7. Write two advantage and disadvantage of **RING** topology. 2

Ans:

Advantages:

1. Short cable length.
2. No wiring closet space required.

Disadvantages:

1. Node failure causes network failure
2. difficult to diagnose faults

8. Define Open Source Software, Free Software, Freeware, and Shareware**.** **2**

Ans:

Free Software : The S/W's is freely accessible and can be freely used changed improved copied and distributed by all and payments are needed to made for free S/W.

Open Source Software : S/w whose source code is available to the customer and it can be modified and redistributed without any limitation .OSS may come free of cost but nominal charges has to pay nominal charges (Support of S/W and development of S/W).

Freeware: Freeware are the software freely available , which permit redistribution but not modification (and their source code is not available). Freeware is distributed in *Binary Form* (ready to run) without any licensing fees.

Shareware: Software for which license fee is payable after some time limit, its source code is not available and modification to the software are not allowed.

9. What is the difference between WAN and MAN? 2

Ans: MAN (Metropolitan Area Network) is the network spread over a city.
WAN (Wide Area Network) spread across countries.

10. What is the purpose of using FTP? 1

Ans: (i)To promote sharing of files (computer programs and/or data).
(ii)To encourage indirect or implicit use of remote computers

11. What is a Modem? 1

Ans: A modem is a computer peripheral that allows you to connect and communicate with other computers via telephone lines.

12. How is a Hacker different from a Cracker? 1

Ans: Hackers are more interested in gaining knowledge about computer systems and possibly using this knowledge for playful pranks.

Crackers are the malicious programmers who break into secure systems

13. Expand the following terms with respect to Networking: 2

(i) Modem (ii) WLL (iii) TCP/IP (iv) FTP

Ans: (i) Modem : Modulator/Demodulator
(ii) WLL: Wireless in Local Loop
(iii) TCP/IP: Transmission Control Protocol/Internet Protocol
(iv) FTP: File Transfer Protocol

14. What are Protocols? 1

Ans: A protocol means the rules that are applicable for a network.
It defines the standardized format for data packets, techniques for detecting and correcting errors and so on.

A protocol is a formal description of message formats and the rules that two or more machines must follow to exchange those messages.
E.g. using library books.

Types of protocols are:

1. HTTP
1. FTP
2. TCP/IP
3. SLIP/PPP

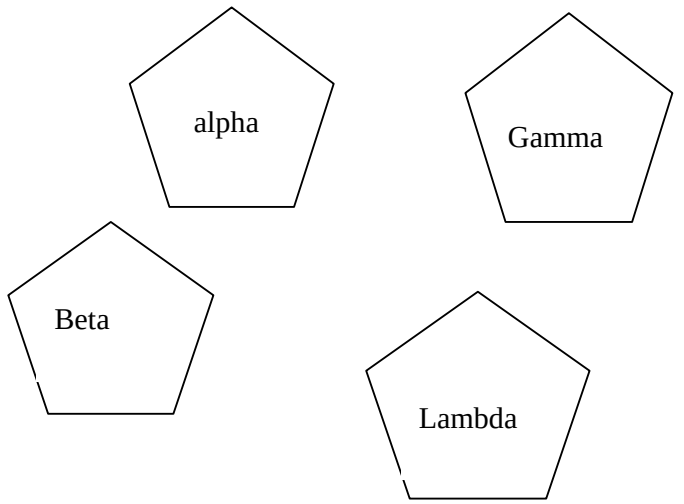
15. What is the difference between Repeater and a Bridge? 1

Ans: A Repeater is a network device that amplifies and restores signals for long distance transmission where as a Bridge is a network device that established an intelligent connection between two local networks with the same standard but with different types of cables.

HOTS (HIGHER ORDER THINKING SKILLS)

4 Marks Questions

1. Knowledge Supplement Organization has set up its new centre at Mangalore for its office and web based activities. It has four building as shown in the diagram below 4



Centre to Centre distance between various buildings

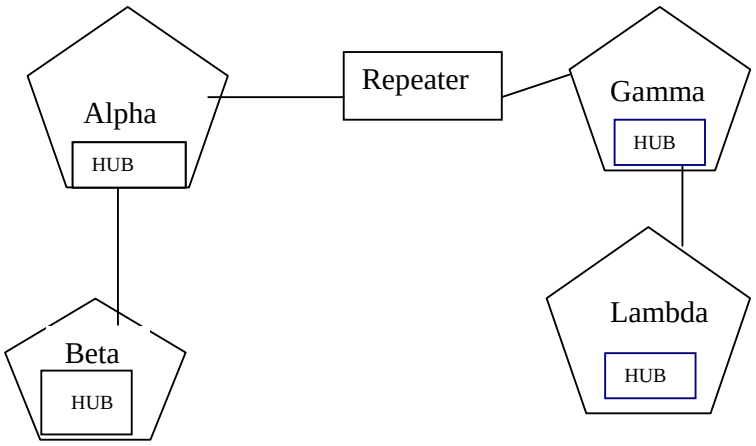
Alpha	25
Beta	50
Gamma	125
Lambda	10

Alpha to Beta	50m
Beta to gamma	150m
Gamma to Lambda	25m
Alpha to Lambda	170m
Beta to Lambda	125m
Alpha to Gamma	90m

(a) Suggesting a cable layout of connection between building state with justification where Server, Repeater and hub will be placed. 2

(b) The organization is planning to link its front office situated in the city in a hilly region where cable connection is not feasible, suggest an economic way to connect it with reasonably high speed? 2

Ans: (i)The most suitable place to house the server of this organization would be building Gamma , as this building contains the maximum number of computers , thus decreasing the cabling cost for most of the computers as well as increasing the efficiency of the maximum computers in the network Distance between alpha to gamma and beta to gamma is large so there repeater will require and hub is necessary for all premises because it is used in local networking.



(ii) The most economic way to connect it with a reasonable high speed would be to use radio wave transmission, as they are easy to install, can travel long distances, and penetrate buildings easily, so they are widely used for communication, both indoors and outdoors. Radio waves also have the advantage of being omni directional, which is they can travel in all the directions from the source, so that the transmitter and receiver do not have to be carefully aligned physically.

2. Software Development Company has set up its new center at Jaipur for its office and web based activities. It has 4 blocks of buildings as shown in the diagram below:

Center to center distances between various blocks

Black A to Block B	50 m
Block B to Block C	150 m
Block C to Block D	25 m
Block A to Block D	170 m
Block B to Block D	125 m
Block A to Block C	90 m

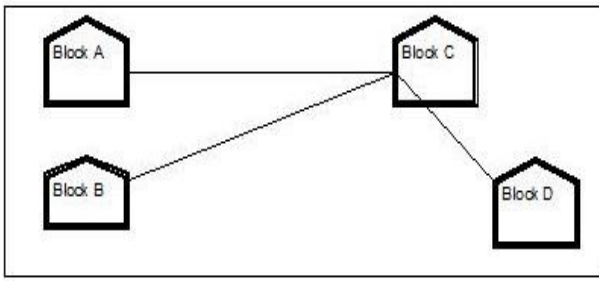
Number of Computers

Black A	25
Block B	50
Block C	125
Block D	10

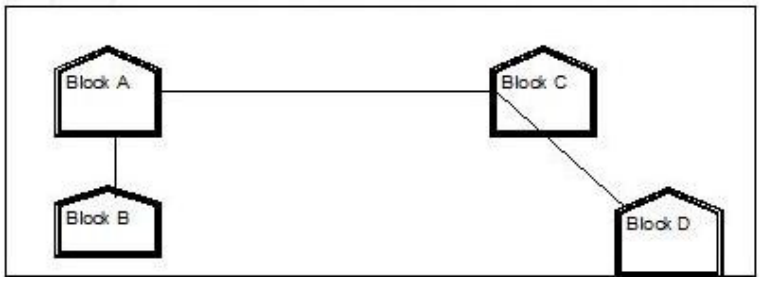
e1) Suggest a cable layout of connections between the blocks.
e2) Suggest the most suitable place (i.e. block) to house the server of this company with a suitable reason.
e3) Suggest the placement of the following devices with justification
(i) Repeater
(ii) Hub/Switch
e4) the company is planning to link its front office situated in the city in a hilly region where cable connection is not feasible, suggest an economic way to connect it with reasonably high speed?

Ans:

(e1) (Any of the following option)
Layout Option 1



Layout Option 2



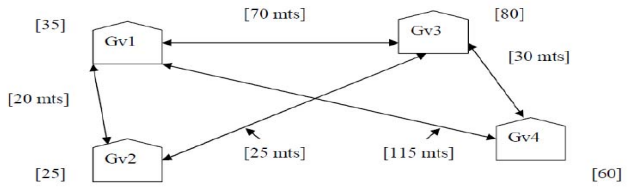
(e2) The most suitable place / block to house the server of this organization would be Block C, as this block contains the maximum number of computers, thus decreasing the cabling cost for most of the computers as well as increasing the efficiency of the maximum computers in the network.

(e3)
(i) For Layout 1, since the cabling distance between Blocks A and C, and that between B and C are quite large, so a repeater each would ideally be needed along their path to avoid loss of signals during the course of data flow in these routes.
For layout 2, since the distance between Blocks A and C is large so a repeater would ideally be placed in between this path.

(ii) In both the layouts, a hub/switch each would be needed in all the blocks, to Interconnect the group of cables from the different computers in each block.

(e4) The most economic way to connect it with a reasonable high speed would be to use radio wave transmission, as they are easy to install, can travel long distances, and penetrate buildings easily, so they are widely used for communication, both indoors and outdoors. Radio waves also have the advantage of being omni directional, which is they can travel in all the directions from the source, so that the transmitter and receiver do not have to be carefully aligned physically.

3. Ram Goods Ltd. has following four buildings in Ahmedabad city.

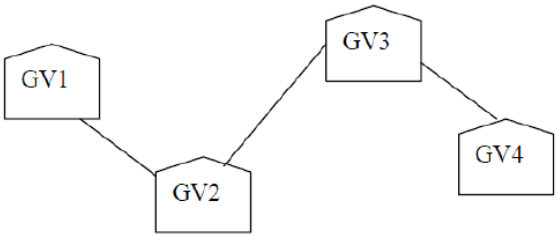


[ ] – Shows computers in each building

→ - Shows distance

Computers in each building are networked but buildings are not networked so far. The company has now decided to connect building also.
(a) Suggest a cable layout for these buildings.
(b) In each of the buildings, the management wants that each LAN segment gets a dedicated bandwidth i.e. bandwidth must not be shared. How can this be achieved?
(c) The company also wants to make available shared Internet access for each of the buildings. How can this be achieved?
(d) The company wants to link its head office in GV1 building to its another office in Japan.
(i) Which type of transmission medium is appropriate for such a link?
(ii) What type of network would this connection result into?

Ans:
(a) Total cable length required for this layout = 75 mts



(b) To give dedicated bandwidth, the computers in each building should be connected via switches as switches offer dedicated bandwidths.

(c) By installing routers in each building, shared internet access can be made Possible.
(d) (i) Satellite as it can connect offices across globe.
(ii) WAN (Wide Area Network)

4. Abhivandan Sanskriti in Udaipur is setting up the network between its different wings. There are 4 wings named as SENIOR(S), MIDDLE(M), JUNIOR(J) and OFFICE(O).
Distance between the various wings are given below:

Wing O to Wing S	100m
Wing O to Wing M	200m
Wing O to Wing J	400m
Wing S to Wing M	300m
Wing S to Wing J	100m
Wing J to Wing M	450m

No. of Computers

Wing O	10
Wing S	200
Wing M	100
Wing J	50

(i) Suggest a suitable Topology for networking the computer ofall wings.
(ii) Name the wing where the server to be installed. Justify your answer.
(iii) Suggest the placement of Hub/Switch in the network.
(iv) Mention an economic technology to provide internet accessibility to all wings.

Ans:
(i) Star or Bus or any other valid topology.
(ii) Wing S, because maximum number of computers are located at Wing S.
(iii) Hub/ Switch in all the wings.
(iv) Coaxial cable/Modem/LAN/TCP-IP/Dialup/DSL/Leased Lines
or any other valid technology.

SOLVED QUESTIONS

INDEX

UNIT 1 : Programming in C++

1 Mark Questions

1. Observe the program segment carefully and answer the question that follows:

```
class member
{
int member_no;
        char member_name[20];
public:
        void enterDetails( );
        void showDetail( );
int getMember_no( ){ return member_no;}
};
void update(member NEW )
{
fstream File;
File.open( "member.dat", ios::binary|ios::in|ios::out) ;
member i;
while(File .read((char*) & i , sizeof (i)))
{
        if(NEW . getMember_no( ) = = i . getMember_no( ))
        {
        File.seekp( _____ , ios::cur ) //Paremeter Missing
        File.write((char*) &NEW , sizeof (NEW));
        }
}
File.close() ;
}
```

If the function update( ) is supposed to modify a record in the file " member.dat" with the values of member NEW passed as argument, write the appropriate parameter for the missing parameter in the above code, so as to modify record at
its proper place.

2. Observe the program segment given below carefully, and answer the question that follows:

```
class Applicant
{
long AId; //Applicant's Id
char Name[20]; //Applicant's Name
float Score; //Applicant's Score
public:
void Enroll();
void Disp();
void MarksScore(); //Function to change Score
long R_Aid() {return Aid;}
};
void ScoreUpdate(long Id)
{
fstream File;
File.open("APPLI.DAT",ios::binary|ios::in|ios::out);
Applicant A;
int Record=0,Found=0;
while (!Found&&File.read((char*)&C, sizeof(c)))
{
if (Id==A.R_Aid())
{
```

```
cout<<"Enter new Score…";
cin>>A.MarksScore();
_____ //statement 1
_____ //statement 2
Found = 1;
}
Record++;
}
if(Found==1) cout<<"Record Updated";
File.close();
}
```

Write the Statement1 to position the File Pointer at the beginning of the Record for which the Applicant's Id matches with the argument passed, and Statement2 to write the updated Record at that position.

3. Observe the program segment carefully and answer the question that follows:

```
class student
{
int student_no;
char student_name[20];
int mark;
public:
void enterDetail( );
void showDetail( );
void change_mark( ); //Function to change the mark
int getStudent_no( ){ return student_no;}
};
void modify( int y )
{
fstream File;
File.open( "student.dat", ios::binary|ios::in|ios::out) ;
student i;
int recordsRead = 0, found = 0;
while(!found && File .read((char*) & i , sizeof (i)))
{
recordsRead++;
if(i . getStudent_no( ) = = y )
{
i . change_mark( );
_____//Missing statement 1
_____//Missing statement 2
found = 1;
}
}
if( found = = 1)
cout<<"Record modified" ;
File.close() ;
}
```

If the function modify( ) is supposed to change the mark of a student having student_no y in the file "student.dat", write the missing statements to modify the student record.

4. Observe the program segment carefully and answer the question that follows:

```
class item
{
int item_no;
char item_name[20];
```

```
public:
void enterDetail( );
void showDetail( );
int getItem_no( ){ return item_no;}
};
void modify(item x, int y )
{
fstream File;
File.open( “item.dat”, ios::binary | ios::in | ios::out) ;
item i;
int recordsRead = 0, found = 0;
while(!found && File.read((char*) &i , sizeof (i)))
{
recordsRead++;
if(i . getItem_no( ) = = y )
{
_____//Missing statement
File.write((char*) &x , sizeof (x));
found = 1;
}
}
if(! found)
cout<<”Record for modification does not exist” ;
File.close() ;
}
```

If the function modify( ) is supposed to modify a record in the file “ item.dat “,which item_no is y, with the values of item x passed as argument, write theappropriate statement for the missing statement using seekp( ) or seekg( ),whichever is needed, in the above code that would write the modified record at its proper place.

5 Observe the program segment carefully and answer the question that follows:

```
class member
{
int member_no;
char member_name[20];
public:
void enterDetail( );
void showDetail( );
int getMember_no( ){ return member_no;}
};
void update(member NEW )
{
fstream File;
File.open( “member.dat”, ios::binary|ios::in|ios::out) ;
member i;
while(File .read((char*) & i , sizeof (i)))
{
if(NEW . getMember_no( ) = = i . getMember_no( ))
{
_____//Missing statement
File.write((char*) &NEW , sizeof (NEW));
}
}
File.close() ;
}
```

If the function update( ) is supposed to modify the member_name field of a record in the file " member.dat" with the values of member NEW passed as argument, write the appropriate statement for the missing statement using seekp( ) or seekg( ), whichever is needed, in the above code that would write the modified record at its proper place.

6. Observe the program segment carefully and answer the question that follows:

```
class item
{
int item_no;
char item_name[20];
public:
void enterDetails( );
void showDetail( );
int getItem_no( ){ return item_no;}
};
void modify(item x )
{
fstream File;
File.open( "item.dat", _____ ) ; //parameter missing
item i;
while(File .read((char*) & i , sizeof (i)))
{
if(x . getItem_no( ) = = i . getItem_no( ))
{
File.seekp(File.tellg( ) – sizeof(i));
File.write((char*) &x , sizeof (x));
}
else
File.write((char*) &i , sizeof (i));
}
File.close() ;
}
```

If the function modify( ) modifies a record in the file " item.dat " with the values of item x passed as argument, write the appropriate parameter for the missing parameter in the above code, so as to modify record at its proper place.

7. Observe the program segment carefully and answer the question that follows:

```
class item
{
int item_no;
char item_name[20];
public:
void enterDetail( );
void showDetail( );
int getItem_no( ){ return item_no;}
};
void modify(item x )
{
fstream File;
File.open( "item.dat", ios::binary|ios::in|ios::out ) ;
item i;
while(File .read((char*) & i , sizeof (i)))//Statement 1
{
if(x . getItem_no( ) = = i . getItem_no( ))
{
File.seekp(File.tellg( ) – sizeof(i));
File.write((char*) &x , sizeof (x));
```

```
}
}
File.close() ;
}
```

If the function modify( ) modifies a record in the file " item.dat" with the values of item x passed as argument, rewrite statement 1 in the above code using eof( ) ,so as to modify record at its proper place.

8.. Observe the program segment given below carefully and fill the blanks marked as Statement 1 and Statement 2 using seekp() and seekg() functions for performing the required task.

```
#include <fstream.h>
class Item
{
int Ino;char Item[20];
public:
//Function to search and display the content from a particular record number
void Search(int );
//Function to modify the content of a particular record number
void Modify(int);
};
void Item::Search(int RecNo)
{
fstream File;
File.open("STOCK.DAT",ios::binary| ios::in);
_____ //Statement 1
File.read((char*)this,sizeof(Item));
cout<<Ino<<"==>"<<Item<<endl;
File.close();
}
void Item::Modify(int RecNo)
{
fstream File;
File.open("STOCK.DAT",ios::binary|ios::in|ios::out);
cout>>Ino;
cin.getline(Item,20);
_____ //Statement 2
File.write((char*)this,sizeof(Item));
File.close();
}
```

9. Observe the program segment given below carefully and fill the blanks marked as Statement 1 and Statement 2 using seekg() and tellg() functions for performing the required task.

```
#include <fstream.h>
class Employee
{
int Eno;char Ename[20];
public:
//Function to count the total number of records
int Countrec();
};
int Item::Countrec()
{
11
fstream File;
File.open("EMP.DAT",ios::binary|ios::in);
_____ //Statement 1- To take the file pointer to
//the end of file.
int Bytes =
```

_____ //Statement 2-To return total number of
bytes from the beginning of
file to the file pointer.
int Count = Bytes / sizeof(Item);
File.close();
return Count;
}

11. A file named as “STUDENT.DAT” contains the student records, i.e. objects of class student.Write the command to open the file to update a student record. (Use suitable stream class and file mode(s).

12. A file named as “EMPLOYEE.DAT” contains the employee records, i.e. objects of class employee.Assuming that the file is just opened through the object FILEof fstream class, in the required file mode, write the command to position the putpointer to point to fifth record from the last record.

13. A file named as “EMPLOYEE.DAT” contains the student records, i.e. objects of class employee. Assuming that the file is just opened through the object FILE of fstream class, in the required File mode, write the command to position the get pointer to point to eighth record from the beginning.

2 Marks Questions
Programming in C++

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

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

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

```
#include <iostream.h>
void main( )
{
int x, sum =0;
cin>>n;
for (x=1;x<100, x+=2)
if x%2=0
sum+=x;
cout<< “sum=” >>sum;
}
```

3. Rewrite the following codes after removing errors, if any, in the following snippet. Explain each error.

```
#include<iostream.h>
void main( )
{
int x[5], *y, z[5]
for (i = 0; i < 5; i ++
{
x[i] = i;
z[i] = i + 3;
y = z;
x = y;
}
```

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

```
#include <iostream.h>
void main( )
{
int x, sum =0;
cin>>n;
for (x=1;x<100, x+=2)
if x%2=0
sum+=x;
cout<< “sum=” >>sum;
}
```

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

```
#include <iostream.h>
void main( )
{
struct Book
```

```
{
char Book_name[20];
char Publisher_name[20];
int Price = 170;
} New Book;
gets(Book_name);
gets(Publisher_name);
}
```

6. Will the following program execute successfully? If no, state the reason(s) :

```
#include<iostream.h>
#include<stdio.h>
#define int M=3;
void main( )
{
const int s1=10;
int s2=100;
char ch;
getchar(ch);
s1=s2*M;
s1+M = s2;
cout<<s1<<s2 ;
}
```

7. Rewrite the following program after removing the syntactical errors (if any). Underline each correction.

```
#include<iostream.h>
void main()
{
char arr{} = {12, 23, 34, 45};
int ptr = arr;
int val = *ptr; cout << *val << endl;
val = *ptr++; cout << val << endl;
val = *ptr : cout << val >> endl;
val = *++ptr; cout << val << endl;
}
```

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

```
#include<iostream.h>
const int dividor 5;
void main( )
{ Number = 15;
for(int Count=1;Count=<5;Count++,Number -= 3)
if(Number % dividor = 0)
{
cout<<Number / Dividor;
cout<<endl;
}
else
cout<<Number + Dividor <<endl;
}
```

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

```
#include<iostream.h>
void main( )
{
First = 10, Second = 30;
Text(First;Second);
Text(Second);
}
void Text(int N1, int N2 = 20)
```

```
{
N1=N1+N2;
count<<N1>>N2;
}
```

10. 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 Numbers[Max];
Numbers = {20,50,10,30,40};
for(Loc=Max-1;Loc>=10;Loc--)
cout>>Numbers[Loc];
}
```

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

```
#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<<end1;
}
else
cout<<Value + Multiple <<endl; }
```

12. Will the following program execute successfully? If not, state the reason(s).

```
#include<stdio.h>
void main( )
{ int s1,s2,num;
s1=s2=0;
for(x=0;x<11;x++)
{
cin<<num;
If(num>0)s1+=num;else s2=/num;
}
cout<<s1<<s2; }
```

13. Identify the errors if any. Also give the reason for errors.

```
#include<iostream.h>
void main()
{
const int i =20;
const int * ptr=&i;
(*ptr)++;
int j=15;
ptr =&j;
}
```

14. Identify the errors if any. Also give the reason for errors.

```
#include<iostream.h>
void main()
{
const int i =20;
const int * const ptr=&i;
```

```
(*ptr)++;
int j=15;
ptr =&j;
}
```

15. Identify errors on the following code segment

```
float c[ ] ={ 1.2,2.2,3.2,56.2};
float *k,*g;
k=c;
g=k+4;
k=k*2;
g=g/2;
cout<<"*k="<<*k<<"*g="<<*g;
```

16.. Write the output of the following program. 2

```
void main( )
{
int x=5,y=5;
cout<<x- -;
cout<<",";
cout<- - x;
cout<<",";
cout<<y- -<<","<<- -y;
}
```

17. Predict the output of the following code:

```
# include<iostream.h>
#include<conio.h>
void main()
{
int arr[] = {12, 23, 34, 45};
int *ptr = arr;
int val = *ptr; cout << val << endl;
val = *ptr++; cout << val << endl;
val = *ptr; cout << val << endl;
val = *++ptr; cout << val << endl;
}
```

18. Find the output of the following code.

```
#include<iostream.h>
#include<conio.h>
void main()
{
int arr[] = {12, 23, 34, 45};
int *ptr = arr;
int val = *ptr; cout << val << endl;
val = *ptr++; cout << val << endl;
val = *ptr; cout << val << endl;
val = *++ptr; cout << val << endl;
val = ++*ptr; cout << val << endl;
}
```

19. Find the output of the following code

```
#include<iostream.h>
#include<conio.h>
void main()
{
int arr[] = {12, 23, 34, 45};
int *ptr = arr;
int val = *ptr; cout << val << endl;
val = (*ptr)++; cout << val << endl;
```

```
val = *ptr; cout << val << endl;
val = *++ptr; cout << val << endl;
}
```

20. Write the output of the following program:

```
#include<iostream.h>
#include<conio.h>
void main( )
{
clrscr( );
int a =32;
int *ptr = &a;
char ch = 'A';
char *cho=&ch;
cho+=a; // it is simply adding the addresses.
*ptr + = ch;
cout<< a << " " <<ch<<endl;
}
```

21. Write the output of the following program:

```
#include<iostream.h>
#include<conio.h>
void main( )
{
clrscr( );
int a =32;
int *ptr = &a;
char ch = 'A';
char *cho=&ch;
*cho+=a; // it is adding the values.
cout<< a << " " <<ch<<endl;
}
```

22. Write a function in C++ to print the count of the word the as an independent word in a text file STORY.TXT. 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.

23. Assume a text file "coordinate.txt" is already created. Using this file create a C++ function to count the number of words having first character capital. Example: Do less Thinking and pay more attention to your heart. Do Less Acquiring and pay more Attention to what you already have. Do Less Complaining and pay more Attention to giving. Do Less criticizing and pay more Attention to Complementing. Do less talking and pay more attention to SILENCE. Output will be : Total words are 16

24. Write a function in C++ to count the number of lines present in a text file "STORY.TXT".
25. Write a function in C++ to count the number of alphabets present in a text file "NOTES.TXT".
26. Write a function in C++ to write the characters entered through the keyboard into the file"myfile.txt", until a '#' character is entered.
27. **Answer the questions (i) and (ii) after going through the following class:**

```
class Seminar
{
int Time;
public:
Seminar() //Function 1
{
Time=30;cout<<"Seminar starts now"<<end1;
}
void Lecture() //Function 2
{
cout<<"Lectures in the seminar on"<<end1;
```

```
}
Seminar(int Duration) //Function 3
{
Time=Duration;cout<<"Seminar starts now"<<end1;
}
~Seminar()
//Function 4
{
cout<<"Vote of thanks"<<end1;
}
};
```

i) In Object Oriented Programming, what is Function 4 referred as and when does it get invoked/ called?
ii) In Object Oriented Programming, which concept is illustrated by Function 1 and Function 3 together? Write an example illustrating the calls for these functions.

28. Answer the questions (i) and (ii) after going through the following program

```
#include<iostream.h>
#include<string.h>
class Bazar
{
char Type[20];
char Product[20];
int Qty;
float Price;
Bazar() //Function 1
{
strcpy (Type,"Electronic");
strcpy (Product,"Calculator");
Qty = 10;
Price=225;
}
public:
void Disp( ) //Function 2
{
cout<<Type<<"-"<<Product<<":"<<Qty
<<"@"<<Price<<endl;
}
};
void main( )
{
Bazar B; //Statement 1
B.Disp(); //Statement 2
}
```

(i) Will Statement 1 initialize all the data members for object B with the values given in the Function 1? (Yes OR No). Justify your answer suggesting the correction(s) to be made in the above code.

(ii) What shall be the possible output when the program gets executed? (Assuming, if required – the suggested correction(s) are made in the program)

29. Given a class as follows:

```
class Match
{
int Time;
int Points;
public:
Match(int y, int p) //Conctructor1
{
```

```
Time=y;
Points =p;
}
Match(Match &M); // Constructor 2
};
```

(i) Create an object, such that it invokes Constructor 1.
(ii) Write complete definition for Constructor 2.

30. Answer the questions (i) and (ii) after going through the following class:

```
class player
{
int health;
int age;
public:
player() { health=7; age=17 } //Constructor1
player(int h, int a) {health =h; age = a ; } //Constructor2
player( player &p) { } //Constructor3
~player() { cout<<"Memory Free"; } //Destructor
};
void main(){
player p1(9,26); //Statement1
player p3 = p1; //Statement3
}
```

(i) When p3 object created specify which constructor invoked and why?
(ii) Write complete definition for Constructor3?

31. Assume that a text file named text1.txt already contains some text written into it, write a function named vowelwords(), that reads the file text1.txt and create a new file named text2.txt, which shall contain only those words from the file text1.txt which don't start with an uppercase vowel(i.e., with 'A','E','I','O','U'). for example if the file text1.txt contains:
Take One Apple And one glass milk daily.
Then the file text2.txt shall contain :
Take one glass milk daily.

32. Assume a text file "Test.TXT" is already created. Using this file, create a function to create three files "LOWER.TXT" which contains all the lowercase vowels and UPPER.TXT" which contains all the uppercase vowels and "DIGIT.TXT" which contains all digits.

33. Write a function in C++ to calculate the average word size in a text file "Report.txt", each word is separated by single space or full stop.

34. Create a function FileLowerShow() in c++ which take file name(text files)as a argument and display its all data into lower case.

3 Marks Questions : Programming in C++

1.What will be the output of the program(Assume all necessary header files are included) :

```
#include<iostream.h>
void print (char * p )
{
p = "pass";
cout<<"value is "<<p<<endl;
}
void main( )
{
char * x = "Best of luck";
print(x);
cout<<"new value is "<<x<<endl;
}
```

2. What will be the output of the following program

```
#include<iostream.h>
#include<ctype.h>
#include<conio.h>
#include<string.h>
void changestring(char text[], int &counter)
{
char *ptr = text;
int length=strlen(text);
for(;counter<length-2;counter+=2, ptr++)
{
*(ptr+counter) = tolower(*(ptr+counter));
}
}
void main()
{
clrscr();
int position = 0;
char message[]= “POINTERS FUN”;
changestring(message, position);
cout<<message<< “@” <<position;
}
```

3. Find the output of the following program :

```
#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;
}
```

4. Write the output of the following program:

```
#include<iostream.h>
#include<conio.h>
int a =3;
void demo(int &x, int y, int *z)
{
a+= x;
y*=a;
*z = a+y;
cout<< a << ""<< x << ""<<y << ""<<z <<endl;
}
void main( )
{
clrscr( );
int a = 2, b =5;
demo(::a,a, &b);
cout<< ::a<< ""<<a<< ""<< b<<endl;
}
```

5. Find the ouptput of the following :

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
#include<string.h>
#include<ctype.h>
void main( )
{
char *Name= "IntRAneT";
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]);
else
Name[x]=Name[x-1];
}
puts(Name);
}
```

6. Give the output of the following program:

```
void main()
{
int x [] = { 10, 20, 30, 40, 50};
int *p, **q, *t;
p = x;
t = x + 1;
q = &t;
cout << *p << "\t" << **q << "\t" << *t++;
}
```

7. What is the output of the following program if all the necessary header files have been included:

```
char *Name= "a ProFile";
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;
```

8. Find the output of the following program:

```
#include<iostream.h>
void main( )
{
int U=10,V=20;
for(int I=1;I<=2;I++)
{
cout<<"[1]"<<U++<<"&"<<V – 5 <<endl;
cout<<"[2]"<<++V<<"&"<<U + 2 <<endl;
}
}
```

9.

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

Outputs:

(i) DEL : CHN : KOL:

(ii) CHN: KOL : CHN:

(iii) KOL : BOM : BNG:

(iv) KOL : CHN : KOL:

10. Find the output of the following program.

```
#include<iostream.h>
void Withdef(int HisNum=30)
{
for(int I=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;
}
```

11. Find the output of the following program:

```
#include<iostream.h>
void main( )
{
long NUM=1234543;
int F=0,S=0;
do
{
int R=NUM % 10;
if (R %2 != 0)
F += R;
else
S += R;
NUM / = 10;
} while (NUM>0);
cout<<F-S;
}
```

12. 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.

```
//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

13. Give the output of the following program:

```
#include<iostream.h>
#include<conio.h>
int g=20;
void func(int &x,int y)
{
x=x-y;
y=x*10;
cout<<x<<','<<y<<'\n';
}
void main( )
{
int g=7;
func(g,::g);
cout<<g<<','<<::g<<'\n';
func(::g,g);
cout<<g<<','<<::g<<'\n';
}
```

14. Find the output of the following program:

```
#include<iostream.h>
struct Box {
int Len, Bre, Hei;
};
void Dimension(Box B)
{
cout << B.Len << " X " << B.Bre << " X ";
cout << B.Hei << endl;
}
void main ( )
{
Box B1 = {10, 20, 8}, B2, B3;
++B1.Hei;
Dimension (B1); //first calling
B3= B1;
++B3.Len;
B3.Bre++;
Dimension (B3); // second function calling
B2= B3;
B2.Hei += 5;
B2.Len - = 2;
Dimension (B2); // third function calling
}
```

15. Find the output of the following program:

```
#include <iostream.h>
struct PLAY
{ int Score, Bonus;
};
void Calculate(PLAY &P, int N=10)
{
P.Score++;P.Bonus+=N; }
void main()
{
PLAY PL={10,15};
Calculate(PL,5);
cout<<PL.Score<<":"<<PL.Bonus<<endl;
Calculate(PL);
cout<<PL.Score<<":"<<PL.Bonus<<endl;
Calculate(PL,15);
cout<<PL.Score<<":"<<PL.Bonus<<endl;
}
```

16.. In the following C++ program , what will the maximum and minimum value of generated with the help of random function.

```
#include<iostream.h>
#include<stdlib.h>
void main()
{
int r;
randomize();
r=random(20)+random(2);
cout<<r;
}
```

17. Study the following program and select the possible output from it:

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

```
const int Max=3;
void main( )
{
randomize();
int Number;
Number=50+random(Max);
for(int P=Number; P >=50;P- -)
cout<<P<<"#";
cout<<endl;
}
```
(i) 53#52#51#50#
(ii) 50#51#52#
(iii) 50#51#
(iv) 51#50#

18. Find the output of the following program:

```
#include<iostream.h>
void main()
{
int A[]={10,20,30,40,50};
int *p=A;
while(*p<30)
{
31
if(*p%3!=0)
*p = *p+2;
else
*p=*p+1;
*p++;
}
for(int J=0;J<=4;J++)
{
cout<<A[J]<< "@";
if(J%3 == 0)
cout<<endl;
}
cout<<A[4]*3<<endl;
}
```

19. Find the output of the following program:

```
#include <iostream.h>
void Changethecontent(int Arr[ ], int Count)
{
for (int C=1;C<Count;C++)
Arr[C-1]+=Arr[C];
}
void main( )
{
int A[ ]={3,4,5},B[ ]={10,20,30,40},C[ ]={900,1200};
Changethecontent(A,3);
Changethecontent(B,4);
Changethecontent(C,2);
for (int L=0;L<3;L++) cout<<A[L]<<'#';
cout<<endl;
for (L=0;L<4;L++) cout<<B[L] <<'#';
cout<<endl;
for (L=0;L<2;L++) cout<<C[L] <<'#'; }
```

20. In the following program, if the value of Guess entered by the user is 65, what will be the expected output(s) from the following options (i), (ii), (iii) and (iv)?

```
#include <iostream.h>
#include <stdlib.h>
void main()
{
int Guess;
randomize();
cin>>Guess;
for (int I=1;I<=4;I++)
{
New=Guess+random(I);
cout<<(char)New;
} }
```

(i) ABBC
(ii) ACBA
(iii) BCDA
(iv) CABD

21.

```
#include <iostream.h>
void Secret(char Str[ ])
{
for (int L=0;Str[L]!='\0';L++);
for (int C=0;C<L/2;C++)
if (Str[C]=='A' || Str[C]=='E')
Str[C]='#';
else
{
char Temp=Str[C];
Str[C]=Str[L-C-1];
Str[L-C-1]=Temp;
}
}
void main()
{
char Message[ ]="ArabSagar";
Secret(Message);
cout<<Message<<endl;
}
```

22. Find the output of the following code.

```
#include<iostream.h>
#include<conio.h>
void main( )
{
clrscr( );
int a =32;
int *ptr = &a;
char ch = 'D';
char *cho=&ch;
*cho+=a;
*ptr += ch;
*ptr *= 3;
ch=ch-30;
cout<< a << "" <<--ch<<endl;
}
```

23. Give the output of the following program.

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

```
void main( )
{
char *p="Difficult";
char c;
c=*p++;
cout<<c<<c++<<++c<<"\n";
char d =c+1;
cout<<d++<<"\n";
cout<<d<<"\n";
cout<<*p;
}
```

24. Given a binary file PHONE.DAT, containing records of the following structure type

```
class Phonlist
{
char Name[20];
char Address[30];
char AreaCode[5];
char PhoneNo[15];
public:
void Register();
void Show();
int CheckCode(char AC[ ])
{
return strcmp(AreaCode,AC);
}
};
```

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

25. Given a binary file TELEPHON.DAT, containing records of the following class Directory:

```
class Directory
{
char Name[20];
char Address[30];
char AreaCode[5];
char Phone_No[15];
public:
void Register();
void Show();
int CheckCode(char AC[])
{
return strcmp(AreaCode,AC[]);
}
};
```

Write a function COPYABC in C++ that would copy only those records having AreaCode as "123" from TELEPHON.DAT to TELEBACK.DAT.

26. Given a binary file SPORTS.DAT, containing records of the following structure type :

```
struct Sports
{
char Event[20];
char Participant[10][30];
};
```

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".

27. Write a function in C++ to search for a BookNo from a binary file “BOOK.DAT”, assuming the binary file is containing the objects of the following class.

```
class BOOK
{
int Bno;
char Title[20];
public:
int RBno(){return Bno;}
void Enter(){cin>>Bno;gets(Title);}
void Display(){cout<<Bno<<Title<<endl;}
};
```

28. Write a function in C++ to add new objects at the bottom of a binary file “STUDENT.DAT”, assuming the binary file is containing the objects of the following class.

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

29. Write a function in C++ to read and display the detail of all the members whose membership type is ‘L’ or ‘M’ from a binary file “CLUB.DAT”. Assuming the binary file “CLUB.DAT” is containing objects of class CLUB, which is defined as follows:

```
class CLUB
{
int Mno.
char Mname[20];
char Type; //Member Type: L Life Member M Monthly member G Guest
public:
void Register( );
void Display( );
char whatType( ) { return type; }
};
```

30. Assuming the class DRINKS defined below, write functions in C++ to perform the following :

(i) write the objects of DRINKS to binary file.

(ii) Read the objects of DRINKS from binary file and display them on screen when Dname has value “ Pepsi”.

```
class DRINKS
{
int DCode;
char DName[13];
int Dsize; // size in litres.
float Dprice;    }
public:
void getdrinks( )
{ cin>>DCode>>DName>>Dsize>>Dprice;}
void showdrinks( )
{ cout<< DCode<<DName<,Dsize<,Dprice;}
char *getname()
{ return Dname;}
};
```

4 Marks Questions : Programming in C++

1. Define a class Travel in C++ with the description given below:

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.

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

```
class CUSTOMER
{
int Cust_no;
char Cust_Name[20];
protected:
void Register();
public:
CUSTOMER();
void Status();
};
class SALESMAN
{
int Salesman_no;
char Salesman_Name[20];
protected:
float Salary;
public:
SALESMAN();
void Enter();
void Show();
};
class SHOP : private CUSTOMER , public SALESMAN
{
```

```
char Voucher_No[10];
char Sales_Date[8];
public:
SHOP();
void Sales_Entry();
void Sales_Detail();
}
```

(iii) Write the names of data members which are accessible from objects belonging to class CUSTOMER.
(iv) Write the names of all the member functions which are accessible from objects belonging to class SALESMAN.
(v) Write the names of all the members which are accessible from member functions of class SHOP.
(iv) How many bytes will be required by an object belonging to SHOP?

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

```
class PUBLISHER
{
char Pub[12];
double Turnover;
protected:
void Register();
public:
PUBLISHER();
void Enter();
void Display();
};
class BRANCH
{
char CITY[20];
protected:
float Employees
public:
BRANCH();
void Haveit();
void Giveit();
};
class AUTHOR : private BRANCH , public PUBLISHER
{
int Acode;
char Aname[20];
float Amount;
public:
AUTHOR();
void Start();
void Show();
};
```

(i) Write the names of data members, which are accessible from objects belonging to class AUTHOR.
(ii) Write the names of all the member functions which are accessible from objects belonging to class BRANCH.
(iii) Write the names of all the members which are accessible from member functions of class AUTHOR.
(iii) How many bytes will be required by an object belonging to class AUTHOR?

4. Define a class TEST in C++ with following description:

Private Members
- TestCode of type integer
- Description of type string
- NoCandidate of type integer

• CenterReqd (number of centers required) of type integer
• A member function CALCNTR() to calculate and return the number of centers as (NoCandidates/100+1)
Public Members
• A function SCHEDULE() to allow user to enter values for TestCode, Description, NoCandidate & call function CALCNTR() to calculate the number of Centres
• A function DISPTEST() to allow user to view the content of all the data members

5. Define a class in C++ with following description:
Private Members
• A data member Flight number of type integer
• A data member Destination of type string
• A data member Distance of type float
• A data member Fuel of type float
• A member function CALFUEL() to calculate the value of Fuel as per the following criteria:
Distance
Fuel
<=1000 500
more than 1000 and <=2000 1100
More than 2000 2200
Public Members
" A function FEEDINFO() to allow user to enter values for Flight Number, Destination, Distance & call function CALFUEL() to calculate the quantity of Fuel
" A function SHOWINFO() to allow user to view the content of all the data members

6. Define a class Clothing in C++ with the following descriptions:
Private Members:
Code of type string
Type of type string
Size of type integer
Material of type string
Price of type float
A function Calc_Price() which calculates and assigns the value of Price as
follows: For the value of Material as “COTTON”
Type Price (Rs.)
TROUSER 1500
SHIRT 1200
For Material other than “COTTON” the above mentioned Price gets reduced by 25%.
Public Members:
A constructor to assign initial values of Code, Type and Material with the word “NOT ASSIGNED” and Size and Price with 0. A function Enter () to input the values of the data members Code, Type, Size and Material and invoke the CalcPrice() function. A function Show () which displays the content of all the data members for a Clothing.

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

```
class Dolls
{
char DCode[5];
protected:
float Price ;
void CalcPrice(float);
public:
Dolls( );
void DInput( );
void DShow( );
};
class SoftDolls: public Dolls
```

```
{
char SDName[20];
float Weight;
public:
SoftDolls( );
void SDInput( );
void SDShow( );
};
class ElectronicDolls: public Dolls
{
char EDName[20];
char BatteryType[10];
int Battieries;
public:
ElectronicDolls ( );
void EDInput( );
void EDShow( );
};
```

(i) Which type of Inheritance is shown in the above example?
(ii) How many bytes will be required by an object of the class ElectronicDolls?
(iii) Write name of all the data members accessible from member functions of the class SoftDolls.
(iv) Write name of all the member functions accessible by an object.

8 consider the following class declaration and answer the question below :

```
class university {
int noc;
protected;
char uname[25];
public:
university();
char state[25];
void enterdata();
void displaydata();
};
class college:public university{
int nod;
char cname[25];
protected:
void affiliation();
public:
college();
void enrol(int ,int);
void show();
};
class department:public college{
char dname[25];
int nof;
public:
department();
void display();
void input();
};
```

(i) Which class's constructor will be called first at the time of declaration of an object of class department?
(ii) How many bytes does an object belonging to class department require?
(iii)Name the member function(s), which are accessed from the object of class department.
(iv) Name the data member, which are accessible from the object of class college.

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

```
class cloth
{
char category[5];
char description[25];
protected:
float price;
public:
void Entercloth( );
void dispcloth( );
};
class Design : protected cloth
{
char design[21];
protected:
float cost_of_cloth;
public:
int design_code;
Design( );
void Enterdesign( );
void dispdesign( );
};
class costing : public cloth
{
float designfee;
float stiching;
float cal_cp( );
protected:
float costprice;
float sellprice;
public:
void Entercost( );
void dispcost( );
costing ( ) { };
};
```

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

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

```
class Regular
{
char SchoolCode[10];
public:
void InRegular( );
void OutRegular( );
};
class Distance
{
char StudyCentreCode[5];
public:
void InDistance( );
void OutDistance( );
};
```

```
class Course : public Regular, private Distance
char Code[5];
float Fees;
int Duration;
public:
void InCourse( );
void OutCourse( );
};
```

(i) Which type of Inheritance is shown in the above example?
(ii) Write names of all the member functions accessible from Outcourse function of class Course.
(iii) Write name of all the members accessible through an object of the Class Course.
(iv) Is the function InRegular( ) accessible inside the function InDistance ( )? Justify your answer.

11. Define a class named ADMISSION in C++ with the following descriptions: Private members:
AD_NO integer (Ranges 10 - 2000)
NAME Array of characters (String)
CLASS Character
FEES Float
Public Members:
- Function Read_Data ( ) to read an object of ADMISSION type
- Function Display( ) to display the details of an object
- Function Draw_Nos ( ) to choose 2 students randomly and display the details.

Use random function to generate admission nos to match with AD_NO.

12. Define a class named MOVIE in C++ with the following description:
Private members
HALL_NO integer
MOVIE_NAME Array of characters (String)
WEEK integer (Total number of weeks the same movie is shown)
WEEK_COLLECTION Float
TOTAL_COLLECTION Float
Public Members
- Function Read_Data( ) to read an object of ADMISSION type
- Function Display( ) to display the details of an object
- Function Update( ) to update the total collection and Weekly collection once in week changes. Total collection will be incremented by Weekly collection and Weekly collection is made Zero

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

```
class Mydata
{
protected:
int data;
public:
void Get_mydata(int);
void Manip_mydata(int);
void Show_mydata(int);
Mydata( );
~Mydata( );
};
class Personal_data
{
protected:
int data1;
public:
void Get_personaldata(int);
void Show_personaldata(int);
Personal_data1( );
```

```
~Personal_data1( );
};
class Person: public Mydata, Personal_data
{
public:
void Show_person(void);
Person( );
~Person( );
};
```

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

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

```
class Book
{
int year_publication;
char title[25];
float price;
public:
Book( );
void input_data( );
void output_data( );
};
class Tape
{
char comp_name[20];
protected:
char comp_addr[35];
public:
Tape( );
void read_data( );
void show_data( );
};
class Publication : private Book , public Tape
{
int no_copies;
public:
Publication( );
void Pub_Entry( );
void Pub_Detail( );
};
```

(i) Write the names of data members which are accessible from objects belonging to class Publication.
(ii) Write the names of all the member functions which are accessible from objects belonging to class Tape.
(iii) Write in which order the constructors will be invoked when an object of class Publication is created .
(iv) How many bytes will be required by an object belonging to class Publication? **15**. Answer the questions (i) to (iv) based on the following code:

```
class vehicle
{
int wheels;
protected:
int passenger;
public:
void inputdata( );
```

```
void outputdata( );
};
class heavyvehicle : protected vehicle
{
int diesel_petrol;
protected:
int load;
public:
void readdata(int, int);
void writedata( );
};
class bus : private heavyvehicle
{
char make[20];
public:
void fetchdata( );
void displaydata( );
};
```

i) Name the base class and derived class of heavyvehicle class.
ii) Name the data member(s) that can be accessed from the function displaydata().
iii) How many bytes will be required by an object of vehicle and heavyvehicle classes respectively?
iv) Is the member function outputdata( ) accessible to the objects of the class heavyvehicle?

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

```
class Animal
{
int leg:
protected:
int tail;
public:
void INPUT (int );
void OUT ( );
};
class wild : private Animal
{
int carniv;
protected:
int teeth;
Public:
void INDATA (int, int )
void OUTDATA( );
};
class pet : public Animal
{
int herbiv;
public:
void Display (void);
};
```

(i) Name the base class and derived class of the class wild.
(ii) Name the data member(s) that can be accessed from function Display ( ).
(iii) Name the member function(s), which can be accessed from the objects of class pet.
(iv) Is the member function OUT ( ) accessible by the objects of the class wild?

17. Answer the questions (i) to (iv) based on the following class declaration:

```
class Medicine
{
```

```
char category[10];
char Date_of_Manufacture[10];
char Date_Of_Expiry[10];
protected:
char company[20];
public:
int x,y;
Medicine( );
void Enter( );
void Show( );
};
class Tablet :protected Medicine
{
protected:
char tablet_name[30];
char volume_label[20];
void disprin( );
public:
float price;
Tablet( );
void enterdet( );
void showdet( );
};
class PainReliever : public Tablet
{
int Dosage_units;
long int tab;
char effects[20];
protected:
int use_within_Days;
public :
PainReliever( );
void enterpr( );
showpr( );
};
```

(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 member functions which are accessible from objects of class PianReliever.
(iv) Write the names of all the data members which are accessible from the functions enterpr().

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

```
class World
{
int H;
protected
int s;
public:
void INPUT(int);
void OUTPUT( );
};
class Country : private World
{
int T;
protected:
int U;
```

```
public :
void INDATA(int, int);
void OUTDATA();   };
class State : public Country
{
int M;
public :
void DISPLAY(void); };
```

(i) Name the base class and derived class of the class Country.
(ii) Name the data member that can be accessed from function DISPLAY( )
(iii) Name the member functions, which can be accessed from the objects of class State.
(iv) Is the member function OUTPUT() accessible by the objects of the class Country ?

ANSWERS: 1 MARK Questions : PROGRAMMING IN C++

1. File.seekp(-sizeof(i), ios::cur);

2. Statement 1:

File.seekp(Record * sizeof(A)); //object name OR File.seekp(Record * sizeof(Applicant)); // class name
File.seekp(File.tellg() - sizeof(A)); OR File.seekp(File.tellg() - sizeof(Applicant));
OR File.seekp(-sizeof(A), ios::cur);
File.seekg(Record * sizeof(A));
OR File.seekg(Record * sizeof(Applicant)); OR File.seekg(-sizeof(A), ios::cur);
Statement 2:
File.write((char*)&A, sizeof(A)); OR File.write((char*)&A, sizeof(Applicant));

3. Same as previous question. In this question it is not defined explicitly that the function of statement1 and statement2 but here first we have to move file pointer at the appropriate place (statement1) and the write the record on that location(statement2)

4.Using seekp()
File.seekp(Record * sizeof(i)); //object name OR File.seekp(Record * sizeof(item)); // class name
OR File.seekp(File.tellg() - sizeof(i)); OR File.seekp(File.tellg() - sizeof(item));
OR File.seekp(-sizeof(i), ios::cur);
Using seekg()
OR File.seekg(Record * sizeof(i)); OR File.seekg(Record * sizeof(item));
OR File.seekg(-sizeof(i), ios::cur);

5. Same as previous question. Just change the class name or object name.
6. You have to complete the syntax of file opening. File.open(“item.dat”, ios::binary|ios::in|ios::out) ;

7. while(!File.eof ( )) ;{
(File .read((char*) & i , sizeof (i));
:
:
8. File.seekg(RecNo*sizeof(Item)); //Statement 1
File.seekp(RecNo*sizeof(Item)); //Statement 2
9. File.seekg(0,ios::end); //Statement 1
File.tellg(); //Statement 2
10. Line1:
File.seekp(Position); OR File.seekp(-sizeof(L), ios::cur);
OR File.seekg(-sizeof(L), ios::cur); OR File.seekg(Position);
Line2:
File.write((char*)&L, sizeof(L)); OR File.write((char*)&L, sizeof(Library));

11. fstream inof(“STUDENT.DAT”,ios::in|ios::out|ios::binary) OR
fstream inof; inof.open(“STUDENT.DAT”,ios::in|ios::out|ios::binary)

12. File.seekp(-5,ios::end);
13. File.seekg(8,ios::beg);

ANSWERS: 2 MARKS : PROGRAMMING IN C++

1.
```
#include <iostream.h>
class MEMBER
{
int Mno;float Fees;
public:
void Register(){cin>>Mno>>Fees;}
void Display(){cout<<Mno<<":"<<Fees<<endl;}
};
void main()
{
MEMBER M;
M.Register();
M.Display();
}
```
2.
```
#include <iostream.h>
struct Pixels
{ int Color,Style;} ;
void ShowPoint(Pixels P)
{ cout<<P.Color<<P.Style<<endl;}
void main()
{
Pixels Point1={5,3};
ShowPoint(Point1);
Pixels Point2=Point1;
Point1.Color+=2;
ShowPoint(Point2);
}
```
3.
```
#include <iostream.h>
void main( )
{
int x, sum =0;
cin >> x ; // x in place of n
for (x=1; x<100 ; x+=2) //semicolon
if x%2= = 0 // double =
sum+=x;
cout<< “sum=” << sum; //<<
}
```
4. In this question first we have to write complete program after removing all the errors.
```
#include<iostream.h>
void main( )
{
int x[5], *y, z[5] ; // semi colon must be used here
for ( int i = 0; i < 5; i ++) // data type of i and closing ) should be there.
{
x[i] = i;
z[i] = i + 3;
*y = z; // wrong assignment ( integer to pointer)
x = *y; // wrong assignment ( Pointer to integer)
}
```

5. Corrected code:

```
#include <iostream.h>
void main()
{
int x, sum =0;
cin >> x ; // x in place of n
for (x=1; x<100 ; x+=2) //semicolon
if x%2= = 0 // double =due to assignment
sum+=x;
cout<< “sum=” << sum; //<<
}
```

6. #include <iostream.h>

```
#include<stdio.h>
void main( )
{
struct Book
{
char Book_name[20];
char Publisher_name[20];
int Price;
} Book New;
gets(New. Book_name);
gets(New. Publisher_name);
}
```

7. #include<iostream.h>

```
#include<stdio.h>
#define int M=3;
void main( )
{
const int s1=10;
int s2=100;
char ch;
getchar(ch);
s1=s2*M;
s1+M = s2;
cout<<s1<<s2 ;
}
```

8. include<iostream.h>

```
void main()
{
int arr[ ] = {12, 23, 34, 45};
int *ptr = arr;
int val = *ptr; cout << val << endl;
val = *ptr++; cout << val << endl;
val = *ptr ; cout << val >> endl;
val = *++ptr; cout << val << endl;
}
```

9. #include<iostream.h>

```
const int dividor= 5;
void main( )
{
int Number = 15;
for(int Count=1;Count<=5;Count++,Number -= 3)
if(Number % dividor = = 0)
{
cout<<Number / Dividor;
```

```
cout<<endl;
}
else
cout<<Number + Dividor <<endl; }
```

10.
```
#include<iostream.h>
void Text(int N1,int N2=20); //Prototype missing
void main( )
{
int First = 10, Second = 30; //Data type missing
Text(First , Second); //Comma to come instead of ;
Text(Second);
}
void Text(int N1, int N2=20)
{
N1=N1+N2;
cout<<N1<<N2; //Output operator << required
}
```

11.
```
#include<iostream.h>
const int Max = 10; //Constant Variable 'Max' must be
//initialized. Declaration Syntax Error
void main( )
{
int Numbers[Max]={20,50,10,30,40};
for(Loc=Max-1;Loc>=0;Loc--)
cout>>Numbers[Loc];
}
```

12.
```
#include<iostream.h>
const int Multiple=3;
void main( )
{
int 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;}
```

13. The program will not execute successfully. Because some syntax errors are there in the program. They are

(i) cin and cout, stream objects used but iostream.h header file is not included in the program.
(ii) x is not declared, it should be declared as int.
(iii) With cin, we should use >> instead of <<.
(iv) The shorthand operator /=, is given wrongly as =/. So the corrected program is as follows:

```
#include<iostream.h>
void main( )
{ int s1,s2,num;
s1=s2=0;
for(int x=0;x<11;x++)
{
cin>>num;
if(num>0)s1+=num;else s2/=num;
}
cout<<s1<<s2; }
```

14.
```
#include<iostream.h>
```

```
void main()
{
const int i=20;
int * ptr=&i;
(*ptr)++; //can not modify a const object
int j=15;
ptr =&j;
}
```

15.
```
#include<iostream.h>
void main()
{
const int i =20;
const int * const ptr=&i; //can not modify a const object
(*ptr)++;
int j=15;
ptr =&j; //can not modify a const pointer
}
```

16. The error statements are

k=k*2; g=g/2; as pointer multiplication and division is not possible.

17. 5,3,4,4

18.
```
int arr[] = {12, 23, 34, 45};
int *ptr = arr;
int val = *ptr; cout << val << endl;
val = *ptr++; cout << val << endl; // first print value then increment the
address(post
// increment in value)
val = *ptr; cout << val << endl; // only values are changing
val = *++ptr; cout << val << endl; // first increment the address and then print the
value.
}
```

Coding only for explanation. Don't write coding in answer sheet. Write only the correct output.

19.
```
int arr[] = {12, 23, 34, 45};
int *ptr = arr;
int val = *ptr; cout << val << endl;
val = *ptr++; cout << val << endl; // first print value then increment the address(post
//increment in value)
val = *ptr; cout << val << endl; // only values are changing
val = *++ptr; cout << val << endl; // first increment the address and then print the
value.
val = ++*ptr; cout << val << endl;
}
```

20.
```
int arr[] = {12, 23, 34, 45};
int *ptr = arr;
int val = *ptr; cout << val << endl;
val = (*ptr)++; cout << val << endl; // (post increment in value)
val = *ptr; cout << val << endl; // only values are changing
val = *++ptr; cout << val << endl; // first increment the address and then print the value.
}
```

21.
```
int arr[] = {2, 33, 44, 55};
int *ptr = arr;
int val = *ptr; cout << val << endl;
val = *++ptr ; cout << val << endl;
val = *ptr; cout << val << endl;
val = * ptr++; cout << val << endl;
}
```

22. 97A

23. The meaning of line *cho+=a is:

*cho= *cho +32
= A+32
=97
='a' (ASCII value of character a)
cho contains the address of ch so ch ='a';
Therefore output would be : 32a

24. 129a

ch =97 (from *cho+=a)
*ptr+=ch
*ptr= *ptr+ch
= 32+ 'a' (Character a)
=32+97
=129
Since *ptr or variable a both are same so variable a =129 and ch = 'a'.

25. void thewordCount()

```
{
ifstream Fil("STORY.TXT");
char String[20];
int C=0;
while(Fil)
{
Fil>>String;
if(strcmpi(String,"the")==0)//case insensitive
C=C+1;
}
cout<<C<<endl;
Fil.close();
}
```

OR

```
void thewordCount()
{
ifstream Fil("STORY.TXT");
char String[20];
int C=0;
while(Fil)
{
Fil>>String;
if(strcmp(String,"the")==0 || strcmp(String,"The")==0)
C=C+1;
}
cout<<C<<endl;
Fil.close();
}
```

OR

```
void thewordCount()
{
ifstream F("STORY.TXT");
char Str[4];
int C=0;
while(F.getline(Str,4,' '))
{
if(strcmp(Str,"the")==0 || strcmp(Str,"The")==0)
C=C+1;
}
```

```
        cout<<C<<endl;
        F.close();
        }
```
26. Hint: Use isupper(word[0])

27.
```
void CountLine()
        {
        ifstream FIL("STORY.TXT");
        int LINES=0;
        char STR[80];
        while (FIL.getline(STR,80))
        LINES++;
        cout<<"No. of Lines:"<<LINES<<endl;
        FIL.close();
        }
```
28.
```
void CountAlphabet()
        {
        ifstream FIL("NOTES.TXT");
        int CALPHA=0;
        char CH=FIL.get();
        while (FIL)
        {
        if (isalpha(CH)) CALPHA++;
        CH=FIL.get();
        }
        cout<<"No. of Alphabets:"<<CALPHA<<endl;
        FIL.close();
        }
```
29.
```
void entercharacter{
        ofstream fout;
        fout.open("string.txt");
        if(!fout) {
        cout<<"\n Unable to open file";
        exit(1);
        }
        char c;
        while((c=cin.get())!='#') // or while((c=getchar())!= '#')
        {
        fout.put(c);
        }
        fout.close();
        }
```
30. i) Destructor, it is invoked as soon as the scope of the object gets over.

ii) Constructor Overloading (or Function Overloading or Polymorphism)

```
Seminar S1; //Function 1
Seminar S2(90); //Function 3
```
(i) No, since the constructor Bazar has been defined in private section or constructor has not been defined in public section. Suggested Correction: Constructor Bazar() to be defined in public

(ii)If the constructor is defined as a public member, the following output shall be generated:

Electronic-Calculator:10@225

32. (i) Match M1(0,0);

(ii) Match (Match &M)
```
{Time=M.Time;
Points=M.Points;}
```
33. (i)When p3 object created , Constructor 3 will be invoked since it is copy constructor.

(ii) complete definition for Constructor 3

```
player( player &p)
{
health = p.health;
age= p.age;}
```

ANSWERS: 3 MARKS :PROGRAMMING IN C++

1. value is pass and New value is Best of luck

2. pOInTErS fUN@10

3. 2@4@8@
4#8#16#20#

4. In function definition, 1 variable by reference, 1 normal variable and 1 pointer variable is used. In calling for a pointer variable, always use & symbol. Calling : demo(3,2,Address of b);
::a and x are having same address (due to by reference)
a =a+x
=3+3=6
y = y * a
= 2*6=12
*z = a + y
= 6+12 = 18
First output : 6(value of a), 6 (value of x), 12, z will be any valid address 0xfff4
Second output: 6,2,18(value stored at the address pointed by z i.e. value of b)

5. iNTTaNEE

6. 10 20 30

Explanation:
here 3 pointer variables p, q and t.
p=x means p contains base address i.e. address of first element of array x. t = x+1 means t contains address of second element i.e. address of second element i.e. address of 20.
q =&t means q contains address of t. cout << *p << "\t" << **q << "\t" << *t++;
In this expression Evaluation will start from right to left.
*t++ means first print the value stored at the address pointed by t i.e. Value of second element i.e. 20. after that the address pointed by t will be incremented 1 int. (2 bytes). Now t will contain the address of 30.Since q contains address of t so *q will give the address of t. therefore **q will give 30.

7. A OROoILE
If *Name = "a ProFIle" then output would be: A OROoHLE

8. [1]10&15
[2]21&13
[1]11&16
[2]22&14

9. Since random(2) gives either 0 or 1, Fly value will be either 1 or 2. (random(n) gives you any number between 0 to n-1)
City[1] is "CHN"
City[2] is "KOL"
Since I value from 0 to 2 (ie<3), 3 iterations will takes place. So the possible output consists 3 strings separated by :, each of them may be either "CHN" or "KOL". So the possible output will be
(ii) CHN : KOL : CHN:
(iv) KOL :CHN : KOL:

10. 20,25,30,
20,25,30,
Number=30

11. 2

12. Expected Output
(ii) 1 2 3 4 5 6 7 8 9 10 11

13. -13,-130
-13,20
33,330
-13,33

14. Execution starts from main( )Output due to first function calling
10X20X9
Due to B3=B1

All the values of B1 will be assigned to B3. Now B3 contains: Length=10, Breadth=20, Height=9 Due to ++B1. Len, Length will be 11. Due to B3. Bre++, Breadth will be 20 (Post Increment) but in next statement incremented value will be used. So second calling will take (11,21,9). Therefore output for second calling:

11X21X9
9X21X14 (Third calling)

Therefore final output will be:

10X20X9
11X21X9
9X21X14

15. 11:20
12:30
13:45

16. Minimum Value: 0 Maximum Value:20

17. 51#50#

18. 12@
22@30@40@
50@150

19. 7#9#5#
30#50#70#40#

20. (i) ABBC

21. #agaSbarr

22. 396 E

23. FEE
G
H
i

24. void TRANSFER()
{
Phonlist P;
fstream fin,fout;
fin.open("PHONE.DAT",ios::binary|ios::in);
fout.open("PHONBACK.DAT",ios::binary|ios::out);
while(fin.read((char*)&P,sizeof(P)))
{
if(P.CheckCode("DEL")==0)
fout.write((char*)&P,sizeof(P));
}
fin.close(); //ignore
fout.close(); //ignore
}
OR
void TRANSFER()
{
Phonlist P;
fstream fin,fout;
fin.open("PHONE.DAT",ios::binary|ios::in);
fout.open("PHONBACK.DAT",ios::binary|ios::in);

```
if(fin)
{
fin.read((char*)&P,sizeof(P));
while(!fin.eof())
{
if(P.CheckCode("DEL")==0)
fout.write((char*)&P,sizeof(P));
fin.read((char*)&P,sizeof(P));
}
}
fin.close(); //ignore
fout.close(); //ignore
}
OR
ifstream fin("PHONE.DAT",ios::binary);
ofstream fout("PHONBACK.DAT",ios::binary);
OR
ifstream fin("PHONE.DAT",ios::binary);
ofstream fout("PHONBACK.DAT",ios::binary);
```

25. //Function to copy records from TELEPHON.DAT to TELEBAC.DAT

```
void COPYABC()
{
fstream IS("TELEPHON.DAT",ios::binary|ios::in);
fstream OA("TELEBACK.DAT",ios::binary|ios::out);
Directory D;
while(IS.read((char*) &D,sizeof(D)))
{
if(D.CheckCode("123")==0)
OA.write((char *)&D,sizeof(D));
}
IS.close();
OA.close();
}
```

26. //Function to copy records from SPORTS.DAT to ATHELETIC.DAT

```
void SP2AT()
{
Sports S;
fstream IS("SPORTS.DAT",ios::binary|ios::in);
fstream OA("ATHLETIC.DAT",ios::binary|ios::out);
while(IS)
{
IS.read((char*) &S,sizeof(S));
if(strcmp(S.Event,"Athletics")==0)
OA.write((char *)&S,sizeof(S));
}
IS.close();
OA.close();
}
OR
void SP2AT()
{
fstream F1,F2;
Sports S;
F1.open("SPORTS.DAT",ios::binary|ios::in);
F2.open("ATHLETIC.DAT",ios::binary|ios::out);
while(F1.read((char*) &S,sizeof(S)))
```

```
        {
        if(!strcmp(S.Event,"Athletics"))
        F2.write((char *)&S,sizeof(S));
        }
        F1.close();
        F2.close();
        }
        OR
        void SP2AT()
        {
        fstream F1,F2;
        Sports S;
        F1.open("SPORTS.DAT",ios::binary|ios::in);
        F2.open("ATHLETIC.DAT",ios::binary|ios::out);
        while(F1)
        {
        F1.read((char*) &S,sizeof(S));
        if(!strcmp(S.Event,"Athletics"))
        F2.write((char *)&S,sizeof(S));
        }
        F1.close();
        F2.close();
        }
27. void BookSearch()
        {
        Fstream FIL("BOOK.DAT", ios::binary|ios::in);
        BOOK B;
        int bn, Found=0;
        cout<<"Enter Book Num to search…";
        cin>>bn;
        while(FIL.read((char*)&S, sizeof(S)))
        if(B.RBno()==bn)
        {
        B.Dispaly( );
        Found++;
        }
        if(Found==0)
        cout<<"Sorry!!! Book not found!!!"<<endl;
        FIL.close();
        }
28. void Addnew()
        {
        fstream FIL("STUDENT.DAT",ios::binary|ios::app);
        STUD S;
        int n;
        cout<<"How many objects do you want to add??";
        cin>>n;
        for(int i=0; i<n;i++)
        {
        S.Enter();
        FIL.write((char*)&S,sizeof(S));
        }
        FIL.close();
        }
29. void DisplayL_M( )
        {
```

```
fstream fin("CLUB.DAT",ios::binary|ios::in);
CLUB C;
while(fin.read((char*) &C,sizeof(C)))
{
if(C.whatType()=='L' || C.whatType()=='M')
C.Display();
}
fin.close();
}
```
OR
```
void DisplayL_M( )
{
fstream fin("CLUB.DAT",ios::binary|ios::in);
CLUB C;
while(fin)
{
fin.read((char*) &C,sizeof(C))
{
if(C.whatType()=='L' || C.whatType()=='M')
C.Display();
}
fin.close();
}
```
30. (i)
```
void write( )
{
DRINK D;
ofstream outf("Drink.DAT",ios::binary);
if(!outf)
cout<<"Unable to open file for writing");
else
{
D.getdrinks( ); // reading values through object
Outf.write((char*)&D; sizeof(D)); // writing object to file using fout.
Outf.close( );
}
}
```
(ii)
```
void read( )
{
DRINK D;
ifstream inf("Drink.DAT",ios::binary);
if(!inf)
cout<<"Unable to open file for reading");
else
{
while(inf)
{ inf.read((char*)&D; sizeof(D)); // reading object from file
if((strcmp(D.getname(), "Pepsi")==0)
{
D.showdrinks( ); // display values through object of class
}
inf.close( );
}
}
}
```
Note: (i) Due to more objects, we will use while.

(ii) To find out how many such records are found, then you may use count variable to count the number of records accordingly.

31. Note: in this question, we have to add new object, so we have to use app mode.

```
void add_obj( )
{
customer C;
ofstream outf (“Customer.dat”, ios::binary||ios::app);
if(!outf)
{cout<<”Error”; exit(0);}
else
{
C.enterdata();
Outf.write((char*)&C, sizeof(C));
Out.close();
}
}
```

32. (i)

```
void write( )
{
DRINK D;
ofstream outf(“Drink.DAT”,ios::binary);
if(!outf)
cout<<”Unable to open file for writing”);
else
{
D.getdrinks( ); // reading values through object
Outf.write((char*)&D; sizeof(D)); // writing object to file using fout.
Outf.close( );
}
}
```

(ii)

```
void read( )
{
DRINK D;
ifstream inf(“Drink.DAT”,ios::binary);
if(!inf)
cout<<”Unable to open file for reading”);
else
{
while(inf)
{
inf.read((char*)&D; sizeof(D)); // reading object from
file
if(D.getcode()==1234)
{
D.showdrinks( ); // display values through object
of class
}
inf.close( );
}
}
```

Note: (i) To find out how many such records are found, then you may use count variable to count the number of records accordingly.

Answers: 4 Marks : Programming in C++

1. 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() //Constructor
{
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;
}
```

2. (i) None of data members are accessible from objects belonging to class AUTHOR.
(ii) Enter(), Show()
(iii) Data members: Voucher_No, Sales_Date, Salary
Member function:Sales_Entry(),Sales_Detail(),Enter(),Show(),Register(), Status()
(iv) 66

3. (i)None of data members are accessible from objects belonging to class 4 AUTHOR.
(ii) Haveit(), Giveit()
(iii) Data members: Employees, Acode, Aname, Amount
Member function: Register(), Enter(), Display(), Haveit(), Giveit(), Start(), Show(),
(iv) 70

4. class TEST

```
{
int TestCode;
char Description[20];
int NoCandidate,CenterReqd;
void CALCNTR();
public:
void SCHEDULE();
void DISPTEST();
};
void TEST::CALCNTR()
{
CenterReqd=(NoCandidate/100 + 1);
}
void TEST::SCHEDULE()
{
cout<<"Test Code :";cin>>TestCode;
cout<<"Description :";gets(Description);
cout<<"Number :";cin>>NoCandidate;
CALCNTR();
}
void TEST::DISPTEST()
{
cout<<"Test Code :"<<TestCode<<endl;
cout<<"Description :"<<Description<<endl;
cout<<"Number :"<<NoCandidate<<endl;;
cout<<"Centres :"<<CenterReqd<<endl;;
}
```

5. class FLIGHT

```
{
int Fno;
char Destination[20];
float Distance, Fuel;
void CALFUEL();
public:
void FEEDINFO();
void FEEDINFO();
void SHOWINFO();
};
void FLIGHT::CALFUEL()
{
if (Distance<=1000)
Fuel=500;
else
if (Distance<=2000)
Fuel=1100;
else
Fuel=2200;
}
void FLIGHT::FEEDINFO()
{
cout<<"Flight No :";cin>>Fno;
cout<<"Destination :";gets(Destination);
cout<<"Distance :";cin>>Distance;
CALFUEL();
}
void FLIGHT::SHOWINFO()
```

```
{
cout<<"Flight No :"<<Fno<<endl;
cout<<"Destination :"<<Destination<<endl;
cout<<"Distance :"<<Distance<<endl;;
cout<<"Fuel :"<<Fuel<<endl;;
}
```

6. class Clothing

```
{
char Code[25];
char Type[25];
int Size ;
char Material[30];
float Price;
public:
Clothing();
void Calc_Price();
void Enter();
void Show();
};
Clothing::Clothing()
{
strcpy(Code,”NOT ASSIGNED”);
strcpy(Type,”NOT ASSIGNED”);
Size=0;
strcpy(Material,”NOT ASSIGNED”);
Price=0;
}
void Clothing::Calc_Price() or void Clothing::CalcPrice()
{
if(strcmp(Type,”TROUSER”)==0 &&
strcmp(Material,”COTTON”)==0)
Price=1500;
else if (strcmp(Type,”SHIRT”)==0 &&
strcmp(Material,”COTTON”)==0)
Price=1200;
else if (strcmp(Type,”TROUSER”)==0 &&
strcmp(Material,”COTTON”)!=0)
Price=1500*0.75;
else if (strcmp(Type,”SHIRT”)==0)&& strcmp(Material,”COTTON”)!=
0)
Price=1200*0.75;
}
void Clothing::Enter()
{
gets(Code); // or cin >> Code;
gets(Type); // or cin >> Type;
cin>>Size;
gets(Material); // or cin >> Material;
Calc_Price(); OR CalcPrice();
}
void Clothing::Show()
{
cout<<Code<<Type<<Size<<Material<<Price<<endl;
}
```

7. (i) Hierarchical Inheritance OR Single Level Inheritance

(ii) 41 bytes
(iii) SDName, Weight, Price
(iv) EDInput(), EDShow(),DInput(), DShow()
Note: Constructor functions ElectronicDolls() & Dolls() to be ignored.

8. (i) Constructor of University Class (Top most Base class)
(ii) 106 bytes
(iii) display(), input(), enrol(int,int), show(), enterdata(), displaydata()
(iv) state

9. (i) None of the data members
(ii) void Enterdesign( ), void dispdesign( )
(iii) price, cost_of_cloth, design_code, designfee, stiching, costprice, sellprice;
(iv) 61 bytes

10. (i)Multiple Inheritance (ii) InCourse( ), InDistance( ), OutDistance( ), InRegular( ),OutRegular( )
(iii)InCourse( ), OutCourse( ), InRegular( ), OutRegular( )
(iv) No, function InRegular( ) is not accessible inside the function InDistance( ), because InRegular( ) is a member of class Regular and InDistance( ) is a member of class Distance, and the class Regular and Distance are two independent classes. OR Yes, Reason is that if we call InRegular( ) inside the function InDistance ( ) in this way only

```
void InDistance( )
{
Regular R;
R. InRegular();
}
```

Unit 2: DATA STRUCTURES : ARRAYS
4 Marks Questions

1 Write a function in C++ which accepts an integer array and its size as arguments and replaces elements having even values with its half and elements having odd values with twice its value

2 Write a function in C++ which accepts an integer array and its size as argument and exchanges the value of first half side elements with the second half side elements of the array.
Example : If an array of eight elements has initial content as 2,4,1,6,7,9,23,10 The function should rearrange the array as 7,9,23,10,2,4,1,6.

3Write a function in c++ to find and display the sum of each row and each column of 2 dimensional array. Use the array and its size as parameters with int as the data type of the array.

4 Write a function in C++, which accepts an integer array and its size as parameters and rearrange the array in reverse. Example if an array of five members initially contains the elements as 6,7,8,13,9,19
Then the function should rearrange the array as 19,9,13,8,7,6

5 Write a function in C++, which accept an integer array and its size as arguments and swap the elements of every even location with its following odd location. Example : if an array of nine elements initially contains the elements as 2,4,1,6,5,7,9,23,10 Then the function should rearrange the array as 4,2,6,1,7,5,23,9,10

6 Write a function in C++ which accepts an integer array and its size as arguments and replaces elements having odd values with thrice and elements having even values with twice its value. Example : If an array of five elements initially contains the elements 3,4,5,16,9
Then the function should rearrange the content of the array as 9,8,15,32,27

7 Write function SORTPOINTS() in c++ to sort an array of structure Game in descending order of points using Bubble Sort Note: Assume the following definition of structure Game

```
struct Game
{
long PNo; // Player Number
char PName[20];
long points;
};
```

8 Write a c++ function to shift all the negative numbers to left and positive number in the right side.

9 Define a function SWPCOL() in C++ to swap (interchange) the first column elements with the last column elements, for a two dimensional array passed as the argument of the function.

Example : if the two dimensional array contains
2 1 4 9
1 3 7 7
5 8 6 3
7 2 1 2
After swapping of the content of 1st and last column, it should be
9 1 4 2
7 3 7 1
3 8 6 5
2 2 1 7

10 Define a function SWPROW() in C++ to swap (interchange) the first row elements with the last row elements, for a two dimensional array passed as the argument of the function.

Example : if the two dimensional array contains
2 1 4 9
1 3 7 7

5 8 6 3
7 2 1 2
After swapping of the content of the array will be
80
7 2 1 2
5 8 6 3
1 3 7 7
2 1 4 9

11 Write a function in C++ to print the product of each column of a 2D integer array passed as the argument of the function Example : if the two dimensional array contains
2 1 4 9
1 3 7 7
5 8 6 3
7 2 1 2
Then the output should appears as
Product of Column1 = 70
Product Column2 = 48
Product of column3= 168
Product of Column4=378

12 Write a function in C++ to print the product of each row of a 2D integer array passed as the argument of the function
Example : if the two dimensional array contains
2 1 4 9
1 3 7 7
5 8 6 3
7 2 1 2
Then the output should appears as
Product of Row1 = 72
Product Row2 = 147
Product of Row3= 720
Product of Row4=28

13. Write a function which accept 2D array of integers and its size as arguments and displays the sum of elements which lie on diagonals.
[Assuming the 2D array to be a square matrix with odd dimension ie 3 x 3 , 4 x 4 etc]
Example of the array content is
5 4 3
6 7 8
1 2 9
Output through the function should be
Diagonal One Sum : 21
Diagonal Two: 11

14. 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.
[Assuming the 2D array to be a square matrix with odd dimension ie 3 x 3 , 5 x 5, 7 x 7 etc]
Example of the array content is
5 4 3
6 7 8
1 2 9
Output through the function should be
Middle row: 6 7 9
Middle Column 4 7 2

15. Write a function in C++ which accepts an integer array and its size as arguments and assign the elements into a two dimensional array of integers in the following format
If the array is 1,2,3,4,5,6 if the array is 1,2,3
The resultant 2D array is The resultant 2D array is
1 2 3 4 5 6 1 2 3

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

16. Write a function in C++ which accepts an integer array and its size as arguments and assign the elements into a two dimensional array of integers in the following format
If the array is 1,2,3,4,5,6 if the array is 1,2,3

The resultant 2D array is	The resultant 2D array is
1 2 3 4 5 6	1 2 3
0 1 2 3 4 5	0 1 2
0 0 1 2 3 4	0 0 1
0 0 0 1 2 3	
0 0 0 0 1 2	
0 0 0 0 0 1	

17. Write a function in C++ which accepts an integer array and its size as arguments and assign the elements into a two dimensional array of integers in the following format

If the array is 1,2,3,4,5,6	if the array is 1,2,3
The resultant 2D array is	The resultant 2D array is
1 0 0 0 0 0	1 0 0
1 2 0 0 0 0	1 2 0
1 2 3 0 0 0	1 2 3
1 2 3 4 0 0	
1 2 3 4 5 0	
1 2 3 4 5 6	

18. Write a user defined function named upperhalf() which takes a 2D array A, with size n rows and n cols as arguments and print the upper half of the matrix

19. Write a user defined function lowerhalf() which takes a 2D array, with size n rows and n cols as argument and prints the lower half of the matrix

20 Write the function to find the largest and second largest number from a two dimensional array. The function should accept the array and its size as argument.

21 Write a function in C++ to merge the contents of two sorted arrays A & B into third array C. Assuming array A is sorted in ascending order, B is sorted in descending order, the resultant array is required to be in ascending order.

Linked List, Stack, Queue : 4 Mark Questions

1 Write a function in C++ to perform a PUSH operation in a dynamically allocated stack considering the following :

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

2 Define function stackpush( ) to insert nodes and stackpop( ) to delete nodes, for a linked list implemented stack having the following structure for each node:

```
struct Node
{ char name[20];
int age;
Node *Link; };
class STACK
{ Node * Top;
public:
STACK( ) { TOP=NULL;}
void stackpush( );
void stackpop( );
~STACK( ); };
```

3 Write a function in C++ to perform a PUSH operation in a dynamically allocated stack considering the following:

```
struct node
{ int x,y;
Node *Link;};
```

4 Write a function in C++ to perform a DELETE operation in a dynamically allocated queue considering the following description:

```
struct Node
{ float U,V;
Node *Link; };
class QUEUE
{
Node *Rear, *Front;
public:
QUEUE( ) { Rear =NULL; Front= NULL;}
void INSERT ( );
void DELETE ( );
~QUEUE ( );
};
```

ANSWERS:Data Structures : ARRAYS : 2 MARKS QUESTIONS

1. If (x[i]%2==0)

```
{
x[i]=x[i]/2;
}
else
{
x[i]=x[i]*2;
}
```

2. The exchanging is happening between the elements indicated by the line
6 7 8 13 9 19
i.e. first element with next element from the middle point, second element with second element from the middle point etc. If i for representing the first element and j for representing the element after the middle point then initially i=0 and j=N/2 and both will increment in each

step(i++ and j++).

```
void ChangeValue(int x[],int N)
{
int temp;
for(int i =0,j=N/2;i<N/2;i++,j++)
{
temp=x[i];
x[i]=x[j];
x[j]=temp;
}
cout<<endl;
for(i=0;i<N;i++)
{
cout<<x[i]<<" ";
}
}
```

3.

AR[0][0] AR[0][1] AR[0][2]
AR[1][0] AR[1][1] AR[1][2]
AR[1][0] AR[2][1] AR[2][2]
CS[0] CS[1] CS[2]

Actually we have to find the Sum of the elements in each row and sum of the elements in each column. Since in the above two dimensional array, three rows are there, so three sum will get and each sum will store in the array RS in the places RS[0], RS[1] and RS[2] respectively. Since column also contains three, three column sum will get and each will store in the array CS in the position CS[0], CS[1] and CS[2] respectively.

```
void FindRowColSum(int x[n][n],int R,int C)
{
int temp,CSum=0,RSum=0;
int CS[10],RS[10]; //for getting the Column sum and row sum
/* Whenever we process a 2D array, we need two for loop. Outer for loop for
processing row and inner for loop for processing the column */.
//********* Finding the Row Sum ************
for(i=0;i<R;i++)
{
for(int j=0;j<C;j++)
{
RSum=RSum+x[i][j]; // Adding the elements in
the same Rows. i denote the
row and its value is not
```

```
changing throughout this loop.
}
RS[i]=RSum; // Assining the Row sum to the array RS
RSum=0; //Making RSum Zero to assign the sum of the
// elements in the next row ;
}
cout<<" Printing the Row sum"<<endl ;
for(i=0;i<R;i++)
{
cout<<RS[i]<<" ";
}
cout<<endl;
// ************* Finding the Column sun ****************
/* Small changes are needed in the above code to find the column sum. They
are underlined */
for(i=0;i<C;i++)
{
for(int j=0;j<R;j++)
{
CSum=CSum+x[j][i]; // Adding the elements in the
// same Columns.
}
CS[i]=CSum; // Assigning the Column sum to the array CS
CSum=0; //Making CSum Zero to assign the sum of
//elements in the next column ;
}
cout<<"Printing the column sum"<<endl;
for(i=0;i<C;i++)
{
cout<<CS[i]<<" ";
}
```

4. 6 7 8 13 9 19

Reversing means swap first element and last element, second first element and second last element and so on.

```
void ReverseArray(int x[], int N)
{
int i,j,temp;
/* i for taking the elements from the beginning onwards, so initialize it with 0
and increment its value(i++). j for taking the elements from the end, so initialize
it with N-1 and decrement its value each time (j--) */
for(i=0,j=N-1;i<N/2;i++,j--)
{
temp=x[i];
x[i]=x[j];
x[j]=temp;
}
cout<<"After reversing the Array elements are"<<endl;
for(i=0;i<N;i++)
{
cout<<x[i]<<" ";
}
}
```

5. In this problem you have to take care of two situation.

1. Even number of elements in the array

6 7 8 13 9 19

2. Odd number of elements in the array

```
6 7 8 13 9 19 25
void ChangeOrder(int x[], int N)
{
int i,j,temp,limit;
if(N%2!=2)
{
limit=N-1;
}
else
{
limit=N;
}
for(i=0;i<limit;i+=2)
{
temp=x[i];
x[i]=x[i+1];
x[i+1]=temp;
}
for(i=0;i<N;i++)
{
cout<<x[i];
}
}
```

6.
```
void ChangeValue(int x[],int N)
{
for(int i=0;i<N;i++)
{
if(x[i]%2!=0) // Checking the element is Odd or not
{
x[i]=x[i]*3;
}
else
{
x[i]=x[i]*2;
}
}
cout<<endl;
for(i=0;i<N;i++)
{
cout<<x[i]<<" ";
}
}
```

Note: If the Question is to make element in the odd position to thrice its value and element in the even position to twice its value, then you have to make only one change in the above program to get the answer ie **If((i+1)%2!=0)** instead **of if(x[i]%2 !=0)**

7. You know bubble sort, in which we are using simple array. Here you are asked to sort structure array base on the element points.If in the normal array say x[10] , you are using x[i], here in the structure array say gm[10], you have to use gm[i].points because you are sorting based on the variable points in the structure Game

```
void SORTPOINTS()
{
Game gm[10];
Game temp;
Cout<<"Enter the details of 10 games"<<endl;
For(int i=0;i<10;i++)
{
```

```
        Cin>>gm[i].PNo;
        Gets(gm[i].PName);
        Cin>>gm[i].points;
        }
// Use the logic for bubble sort. Points to note in bubble sort
1. Compare with the adjacent elements ie j and j+1
2. Bigger element goes to the top because the elements in the descending order.
3. Each iteration the smaller elements comes in the bottom.*/
for(i=0;i<n;i++)
{
for(j=0;j<(n-1)-i;j++) // j< (N-1)-i , subtracting i to avoid the
// last elements which are in the correct order  after each loop execution.
        {
        If(gm[j] .points <gm[j+].points)
        {
        temp=gm[j];
        gm[j]=gm[j+1];
        gm[j+1]=temp;
        }
        }
        }
```

8.

1.Take each element of the array from the beginning. say x[n]
2. Check x[n] is -ve or not. If it is negative number do the following steps
2. Check elements from the left side of x[n] and shift the left side element to next right position if it is +ve number..
3. Repeat step 2 till you gets a negative number or till reach the left end of the array.
4. Insert x[n].

```
        void arrangeNegPosNumbers(int x[], int N)
        {
        for(i=0;i<5;i++)
        {
        temp=x[i]; // Taking each element in the array to
        //check +ve or -ve
        if(x[i]<0) // Checking it is -ve
        {
        j=i-1;
        while(j>=0)
        {
        if(x[j]<0)
        {
        break;
        }
        else
        {
        x[j+1]=x[j]; // Shift +ve number
        // to right
        j--;
        }
        }
        x[j+1]=temp; // Inserting the –ve number
        }
        }
        }
```

9. void SWAPCOL(int x[n][n],int R,int C)

```
{
for(i=0;i<R;i++) // for each row
{
Int temp;
for(int j=0,k=C-1;j<C/2;j++,k--)
/* j for taking the elements from the first columns and k for taking the
elements from the last columns. So the initially j=0 and k=C-1 and j will
increase (j++) and k will decrease (k--) * in each step/
{
temp=x[i][j]; //Swap elements in the first and last
// columns
x[i][j]=x[i][k];
x[i][k]=temp;
}
}
for(i=0;i<R;i++)
{
for(int j=0;j<C;j++)
{
cout<<x[i][j]<<" ";
}
cout<<endl;
}
}
```

10. Only small changes have to make compare to the previous question , which are underlined and blackened.

```
void SWAPROW(int x[n][n], int R,intC)
{
int temp;
for(i=0;i<C;i++)
{
for(int j=0,k=R-1;j<R/2;j++,k--)
{
temp=x[j][i];
x[j][i]=x[k][i];
92
x[k][i]=temp;
}
}
for(i=0;i<R;i++)
{
for(int j=0;j<C;j++)
{
cout<<x[i][j]<<" ";
}
cout<<endl;
}
}
```

11.

```
void FindColProd(int x[n][n],int R,int C)
{
int Cprod;
for(i=0;i<C;i++)
{
CProd=1;
for(int j=0;j<R;j++)
```

```
{
CProd= CProd * x[j][i];
}
cout<<"Product of the Column "<<i+1<<" is"<<CProd<<endl;
}
}
```

12.
```
void FindRowProd(int x[n][n],int R,int C)
{
int Rprod;
for(i=0;i<R;i++)
{
RProd=1;
for(int j=0;j<C;j++)
{
RProd= RProd * x[i][j];
}
cout<<"Product of the Row "<<i+1<<" is"<<RProd<<endl;
}
}
```

13. In first diagonal elements both row index and column index is same. In second diagonal element row index is increasing and column index is decreasing.

```
void findDiagonalSum(int x[n][n],int R,int C)
{
// Find the sum of the First diagonal numbers
for(i=0;i<R;i++)
{
Sum1=Sum1+x[i][i]; // Row and Column are represented
// by i itself
}
// Find the sum of the Second diagonal elements
int j=C-1;
for(i=0;i<R;i++)
{
Sum2= Sum2+x[i][j];
j--;
}
cout<<endl;
cout<<Sum1<<endl;
cout<<Sum2;
}
```

14. The row number and Column number is odd numbers and both are same The index of the middle row / column element is Row/2 or Column/2

```
void FindMiddle(int x[n][n], int size)    // size represent No of rows and columns, since both are same.
{
// Find the element in the middle row
int j=size/2;
// Middle Row
for(i=0;i<size;i++)
{
cout<<x[j][i]<<” “;
}
// Middle Column
for(i=0;i<size;i++)
{
```

```
cout<<x[i][j]<<" ";
}
}
```

Note If the question is to find the sum of the middle row and sum of the middle column, then the program should write as given below

```
void FindMiddleSum(int x[n][n], int size)
{
int Sum1=0,Sum2=0;
int j=size/2;
for(i=0;i<size;i++)
{
Sum1=Sum1+x[j][i];
Sum2=Sum2+x[i][j];
}
cout<<endl;
cout<<Sum1<<" "<<Sum2;
}
```

15. We are provided with 1D array and to make 2D array with the values provided in the ID array. By analyzing the 2D array(output) we will get the logic that for the position (i+j)< size of the 2D array, elements are taken from the ID array and all other position value is zero. So the program is as below

```
void Change2Darray(int x[],int size)
for(i=0;i<size;i++)
{
for(int j=0;j<size;j++)
{
if(i+j <size)
{
y[i][j]=x[j];
}
else
{
y[i][j]=0;
}
}
}
for(i=0;i<size;i++)
{
for(int j=0;j<size;j++)
{
cout<<y[i][j]<<" ";
}
cout<<endl;
}
}
```

16. Condition for putting the value is the position **(i<=j) of 2D array** otherwise put zero

```
void Change2Darray(int x[],int size)
{
for(i=0;i<size;i++)
{
int k=0;
for(int j=0;j< size;j++)
{
if(i<=j)
{
y[i][j]=x[k];
```

```
k++;
}
else
{
y[i][j]=0;
}
}
}
for(i=0;i< size;i++)
{
for(int j=0;j< size;j++)
{
cout<<y[i][j]<<" ";
}
cout<<endl;
}
}
```

17. Condition for putting the value is the position **(i>=j) of 2D array** otherwise put zero

```
void Change2Darray(int x[],int size)
{
for(i=0;i<size;i++)
{
for(int j=0;j< size;j++)
{
if(i>=j)
{
y[i][j]=x[j];
}
else
{
y[i][j]=0;
}
}
}
for(i=0;i< size;i++)
{
for(int j=0;j< size;j++)
{
cout<<y[i][j]<<" ";
}
cout<<endl;
}
}
```

Note **All the above three cases only the condition is changing, which is darken and underlined.**

18.

```
void upperhalf(int x[n][n],int R,int C)
{
for(i=0;i<R;i++)
{
for(int j=0;j<C;j++)
{
if(i<=j)
{
cout<<x[i][j];
}
else
```

```
{
cout<<" ";
}
}
cout<<endl;
}
}
```

19.
```
void lowerhalf(int x[n]][n], int R,int C)
for(i=0;i<R;i++)
{
for(int j=0;j<C;j++)
{
if(i>=j)
{
cout<<x[i][j];
}
else
{
cout<<" ";
}
}
cout<<endl;
}
```

20.
```
void findLargestSecondLargest(int x[n][n], int R,int C)
{
int mx,s_,max;
max=x[0][0]; // Assuming x[0][0] as the largest element
s_max=x[0][0]; // Assuming x[0][0] as the Second largest element
for(i=0;i<R;i++)
{
for(int j=0;j<C;j++)
{
if(max<x[i][j]) // if element in the array (x[i][j]) is
// greater than content in the
//variable max
{
s_max=max; // content of the max will
// become the second largest element
max=x[i][j]; // x[i][j] becomes the largest element
}
else if(max>x[i][j] && s_max<x[i][j])
//if the element in the array is less than content of max but greater than s_mx
s_max=x[i][j]; // x[i][j] becomes the second largest
// element
}
}
}
cout<<endl;
cout<<"Maximun Number"<<max<<endl;
cout<<"Second Max Number"<<s_max<<endl;
}
```

21.
```
void AddNSave(int A[],int B[],int C[],int N,int M, int &K)
{
int I=0,J=M-1;
K=0;
while (I<N && J>=0)
```

```
{
if (A[I]<B[J])
C[K++]=A[I++];
else
if (A[I]>B[J])
C[K++]=B[J--];
else
{
C[K++]=A[I++];
J--;
}
}
for (int T=I;T<N;T++)
C[K++]=A[T];
for (T=J;T>=0;T--)
C[K++]=B[T];
}
```

Answers : Linked List, Stack, Queue: 4 Mark Questions

1. Insert new node as the first element

```
void PUSH()
{ Node *newptr=new Node;
cout<<"Enter the informations"<<endl;
cin>>newptr->X;
cin>>newptr->Y;
if(top==NULL) // Linked list is empty
{ Top=newptr; /* New node is the first node*/ }
else
{ newptr->Link=Top;
Top=newptr; } }
```

2. Stack**push( )** is already done // Write yourself

```
void stackpop( ) // Pop from the beginning
{ Node *temp;
if(top==NULL)
{ cout<<"UNDERFLOW……."; }
else
{ temp=Top;
Top=Top->Link;
delete temp; } }
```

3. Insert at the beginning

```
void PUSH()
{ Node *newptr=new Node;
cin>>newptr->x;
cin>>newptr->y;
if(top==NULL) // Stack is empty
{ Top=newptr; }
else
{ newptr->Link=start; // new node will point to the first node;
top=newptr; /* New node becomes to the first node*/ } }
```

4. Hint : delete node from the beginning

```
void DELETE()
{
Node *temp;
if(front==NULL) // No element in the queue
{
cout<<"UNDERFLOW……..";
}
else
{ temp=front;
front=front->Link; // Making the second node as the first one
delete temp; // deleting the previous first node.
}
 }
```

Unit 3 : Database and SQL : 1 mark questions

Q1 Write the expression in relational algebra to :

i. Show the tuples from PRODUCT table where cost of the product is more than 5000.
ii. Show the tuples from PRODUCT table where product_name is 'TV'.
iii. Show the tuples pertaining to prices between 55 and 100 from the table Items.
iv. Show the tuples whose price is more than 55 or qty<10 from the table Items.
v. Show the supplier_name, city where price is more than 1000 from the table Items.

Q2 Write the purpose of following relational algebra statements:

i. _ price>50 (PRODUCTS).
ii. _ city='Chennai' (PRODUCTS)
iii. _ price>20 ^ price <45(SALES)

Q3. Define the terms:

i. Database Abstraction
ii. Data inconsistency
iii. Conceptual level of database implementation/abstraction
iv. Primary Key
v. Candidate Key
vi. Relational Algebra
vii. Domain
viii. Projection

6/8 Marks Questions SQL

1 Write SQL commands for (i) to (viii) on the basis of relations given below:

BOOKS

book_id	Book_name	author_name	Publishers	Price	Type	qty
k0001	Let us C	Sanjay mukharjee	EPB	450	Comp	15
p0001	Genuine	J. Mukhi	FIRST PUBL.	755	Fiction	24
m0001	Mastering c++	Kanetkar	EPB	165	Comp	60
n0002	Vc++ advance	P. Purohit	TDH	250	Comp	45
k0002	Near to heart	Sanjeev	FIRST PUBL.	350	Fiction	30

ISSUED

i. To show the books of FIRST PUBL Publishers written by P.Purohit.
ii. To display cost of all the books written for FIRST PUBL.
iii.Depreciate the price of all books of EPB publishers by 5%.
iv.To display the BOOK_NAME,price of the books whose more than 3 copies have been issued.
v.To show total cost of books of each type.
vi.To show the detail of the most costly book.

2. Write SQL commands for (a) to (f) and write output for (g) on the basis of PRODUCTS relation given below:

PCODE	PNAME	COMPANY	PRICE	STOCK	MANUFACTURE	WARRANTY
P001	TV	BPL	10000	200	12-JAN-2008	3
P002	TV	SONY	12000	150	23-MAR-2007	4
P003	PC	LENOVO	39000	100	09-APR-2008	2
P004	PC	COMPAQ	38000	120	20-JUN-2009	2
P005	HANDYCAM	SONY	18000	250	23-MAR-2007	3

a) To show details of all the PC with stock more than 110.
b) To list the company which gives warranty for more than 2 years.
c) To find stock value of the BPL company where stock value is sum of the products of price and stock.
d) To show number of products from each company.
e) To count the number of PRODUCTS which shall be out of warranty on 20-NOV-2010.
f) To show the PRODUCT name which are within warranty as on date.
g). Give the output of following statement.

(i) Select COUNT(distinct company) from PRODUCT.
(ii) Select MAX(price)from PRODUCT where WARRANTY<=3

Answers: 1 mark questions

Q1 Write the expression in relational algebra to :
vi. Show the tuples from PRODUCT table where cost of the product is more than 5000.
Ans: _ cost>5000(PRODUCTS)
vii. Show the tuples from PRODUCT table where product_name is 'TV'.
Ans: _ product_name='TV'(PRODUCTS)
viii. Show the tuples pertaining to prices between 55 and 100 from the table Items.
Ans: _ price>55^price<100(Items) . Assuming 55 and 100 not included. _ price>=55^price<=100(Items). Assuming 55 and 100 included.
ix. Show the tuples whose price is more than 55 or qty<10 from the table Items.
Ans : _ price>=55 v qty<10(Items)
x. Show the supplier_name, city where price is more than 1000 from the table Items.
Ans : _ supplier_name,city (_ price>1000(Items))

Q2 Write the purpose of following relational algebra statements:
iv. _ price>50 (PRODUCTS).
Ans: To select/show those rows from the table PRODUCTS whose price is more than 50.
v. _ city='Chennai' (PRODUCTS)
Ans: To select/show those rows from the table PRODUCTS where city is Chennai.
vi. _ price>20 ^ price <45(SALES)
Ans: To select/show those rows from the table SALES where city is Chennai.

Q3. Define the terms:
ix. Database Abstraction
Ans: Database system provides the users only that much information that is required by them, and hides certain details like, how the data is stored and maintained in database at hardware level. This concept/process is Database abstraction.
x. Data inconsistency
Ans: When two or more entries about the same data do not agree i.e. when one of them stores the updated information and the other does not, it results in data inconsistency in the database.
xi. Conceptual level of database implementation/abstraction
Ans: It describes what data are actually stored in the database. It also describes the relationships existing among data. At this level the database is described logically in terms of simple data-structures.
xii. Primary Key
Ans : It is a key/attribute or a set of attributes that can uniquely identify tuples within the relation.
xiii. Candidate Key
Ans : All attributes combinations inside a relation that can serve as primary key are candidate key as they are candidates for being as a primary key or a part of it.
xiv. Relational Algebra
Ans : It is the collections of rules and operations on relations(tables). The various operations are selection, projection, Cartesian product, union, set difference and intersection, and joining of relations.
xv. Domain
Ans : it is the pool or collection of data from which the actual values appearing in a given column are drawn.
xvi. Projection
Ans : It is the operation yielding a vertical subset of a given relation , i.e. data under specified columns , in contrast to the horizontal subset(rows) returned by a select operation.

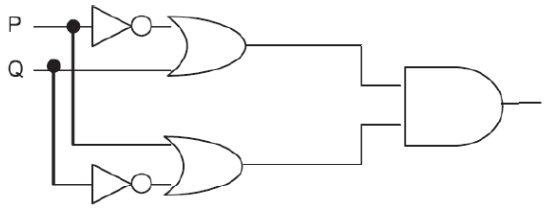
Answers: 6/8 Marks Questions:

1.
i. To show the books of FIRST PUBL Publishers written by P.Purohit.
Ans: select * from books where publishers='FIRST PUBL'
ii. To display cost of all the books written for FIRST PUBL.
Ans: select sum(price*qty) from books where publishers=' FIRST PUBL';
iii.Depreciate the price of all books of EPB publishers by 5%.
Ans: update books set price=price-0.5*price where publishers='EPB';
iv.To display the BOOK_NAME,price of the books whose more than 3 copies have been issued.
Ans: select BOOK_NAME,price from books, issued where
books.book_id=issued.book_id and quantity_issued>3;
v.To show total cost of books of each type.
Ans: select sum(price*qty) from books group by type;
vi.To show the detail of the most costly book.
Ans: select * from books where book_id=(select book_id from books where price=select
max(price) from books));

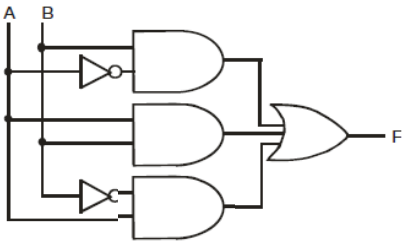
2.
a) To show details of all the PC with stock more than 110.
Ans: select * from products where pname='TV' and stock>110;
b) To list the company which gives warranty for more than 2 years.
Ans: select company from products where warranty>2;
c) To find stock value of the BPL company where stock value is sum of the products of price and stock.
Ans: select sum(price*stock) from PRODUCTS where company='BPL';
d) To show number of products from each company.
Ans: select company,COUNT(*) from products group by company;
e) To count the number of PRODUCTS which shall be out of warranty on 20-NOV-2010.
Ans: select count(*) from products where (20-NOV-2010- manufacture)/365>warranty;
f) To show the PRODUCT name which are within warranty as on date.
Ans: select pname from products where (sysdate- manufacture)/365<warranty;
g). Give the output of following statement.
(i) Select COUNT(distinct company) from PRODUCT.
Ans: 4
(ii) Select MAX(price)from PRODUCT where WARRANTY<=3
Ans: 39000

UNIT 4 : Boolean Algebra : 2 Marks Questions

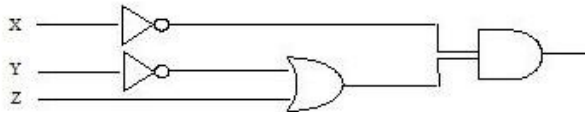
1. Write the equivalent Boolean Expression for the following Logic Circuit



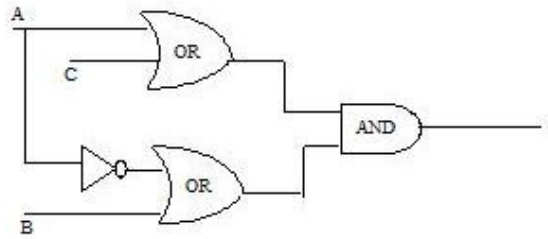
2. Write the equivalent Boolean Expression F for the following circuit diagram :



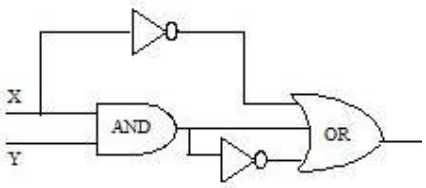
3. Write the equivalent Boolean Expression F for the following circuit diagram :



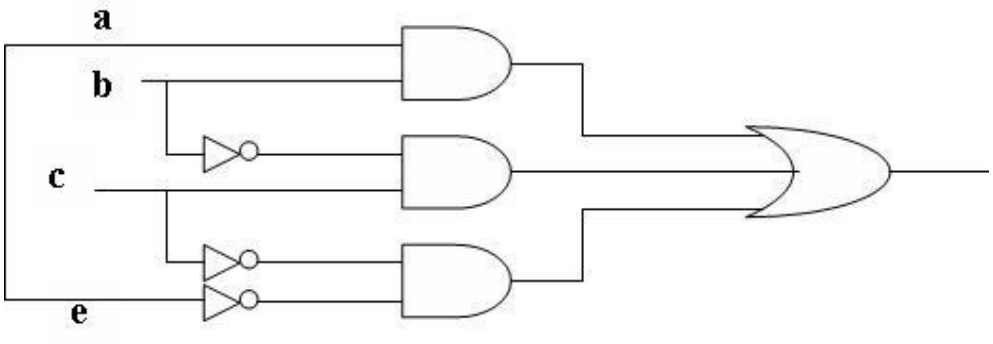
4. Convert the following Boolean expression into its equivalent Canonical Sum of Product Form((SOP)
(X'+Y+Z').(X'+Y+Z).(X'+Y'+Z).(X'+Y'+Z')
5. Convert the following Boolean expression into its equivalent Canonical Product of Sum form (POS):
A.B'.C + A'.B.C +A'.B.C'
6. Draw a Logical Circuit Diagram for the following Boolean Expression:
A.(B+C')
7. Write the equivalent Boolean Expression F for the following circuit diagram :



8. Prove that XY+YZ+YZ'=Y algebraically
9. Express the F(X,Z)=X+X'Z into canonical SOP form.
10. Write the equivalent Boolean Expression for the following Logic Circuit.



11. Interpret the following logical circuit as Boolean expression



12. Design (A+B).(C+D) using NAND Gate
13. Simplify the following Boolean Expression using Boolean postulates and laws of Boolean Algebra.
Z=(a'+a).b'.c+a.b'.c'+a.b.(c+c')
14. Prove x'.y'+y.z = x'yz+x'yz'+xyz+x'yz algebraically.
15. Prove that (a'+b')(a'+b)(a+b')=a'b'.
16. A Boolean function F defined on three input variable X,Y,Z is 1 if and only if the number of 1(One) input is odd (e.g. F is 1 if X=1,Y=0,Z=0). Draw the truth table for the above function and express it in canonical sum of product form.

3 Marks Questions : Boolean Algebra

1. If F(a,b,c,d)=_(0,2,4,5,7,8,10,12,13,15), obtain the simplified form using K-Map.
2. If F(a,b,c,d) = _ (0,1,3,4,5,7,8,9,11,12,13,15), obtain the simplified form using KMap
3. Obtain a simplified form for a boolean expression
F(U,V,W,Z)= _ (0,1,3,5,6,7,10,14,15)
4. Reduce the following boolean expression using K-Map
F(A,B,C,D) = _ (5,6,7,8,9,12,13,14,15)
5. Reduce the following Boolean expression using K-Map:
F(A,B,C,D)=_(0,1,2,4,5,8,9,10,11)
6. Reduce the following Boolean expression using K – Map
F (A, B, C, D) =Σ (0,2,3,4,6,7,8,10,12)
7. Reduce the following Boolean Expression using K-Map:
F(A,B,C,D)= _(0,1,2,4,5,6,8,10)

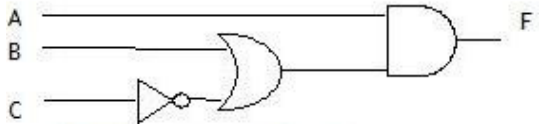
Answers: 2 Marks Questions : Boolean Algebra

1. *F(P,Q)=(P'+Q).(P+Q')*
2. *A'B+AB+AB'*
3. *X'(Y'+Z)*
4.

F(X , Y , Z) = ∏ (4 , 5 , 6 , 7)
= Σ (0 , 1 , 2 , 3)
= X'. Y'. Z' + X'. Y'. Z + X'. Y. Z' + X'. Y. Z

5. *A.B'.C + A'.B.C +A'.B.C' = _(0,1,4,6,7) OR =(A+B+C).(A+B+C').(A'+B+C).(A'+B'+C).(A'+B'+C')*

6.



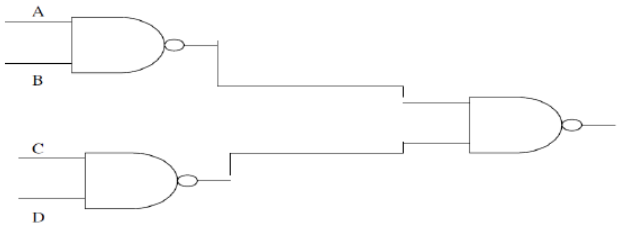
7. *(A+C)(A'+B)*
8. *XY+YZ+YZ'=Y*

L.H.S.
XY+YZ+YZ'
= XY+Y(Z+Z')
=XY+Y=Y(X+1)
=Y.1
=Y=RHS

9. *F(X,Z)=X+X'Z =X(Y+Y')+X'(Y+Y')Z*

=XY+XY'+X'YZ+X'Y'Z
=XY(Z+Z')+XY'(Z+Z')+X'YZ+X'Y'Z
=XYZ+XYZ'+XY'Z+XY'Z'+X'YZ+X'Y'Z

10. *XY+(XY)'+X'*
11. *ab+b'c+c'e'*
12.



13. *Z=(a'+a).b'.c+a.b'.c'+a.b.(c+c')*

RHS=(a'+a).b'.c+a.b'.c'+a.b.(c+c')
=a'bc+ab'c+ab'c'+ab.1
=a'bc+ab'c'+ab'c
=ab'(c+c')+ab'c
=ab'+ab'c
=ab'(1+c)
=ab'

14. *L.H.S.= x'y +y.z*

=x'y.1+1.y.z =x'y(z+z')+(x+x')y.z
=x'yz+x'yz'+xyz+x'yz =RHS

15. *LHS=(a'+b')(a'+b)(a+b')*

=(a'a'+a'b+a'b'+b'b)(a+b')
=(a'+a'b+a'b'+0)(a+b')
=aa'+a'b'+aa'b+a'bb'+a'ab'+a'b'b'
=0+a'b'+0+)+0+0+a'b'=a'b'=RHS

16.

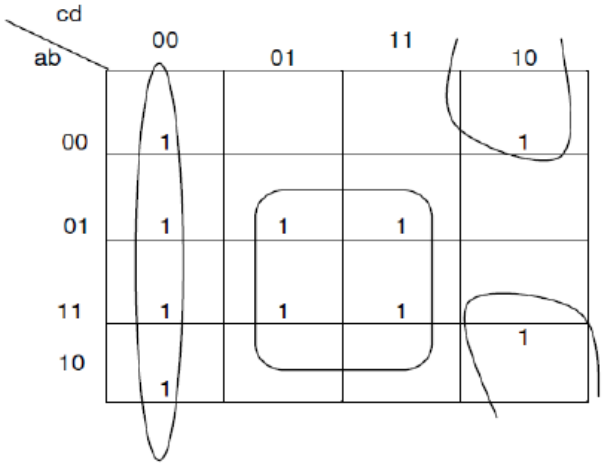
X Y Z F
0 0 0 0
0 0 1 1
0 1 0 1
0 1 1 0
1 0 0 1
1 0 1 0
1 1 0 0
1 1 1 1

Canonical SOP

XYZ'+XY'Z+XY'Z'+XYZ

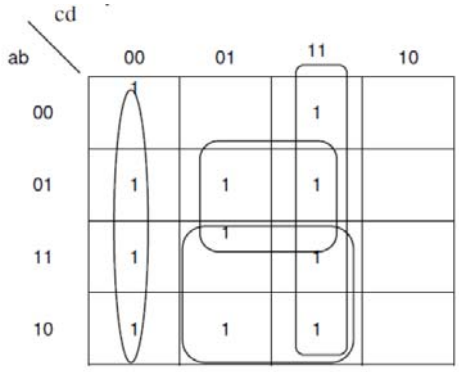
Answers: 3 marks Questions
Boolean Algebra

1.

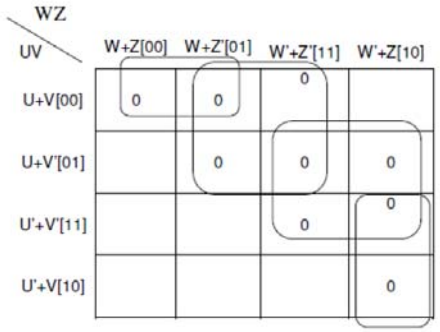


The Final Expression is F=c'd'+bd+b'c'd

2.



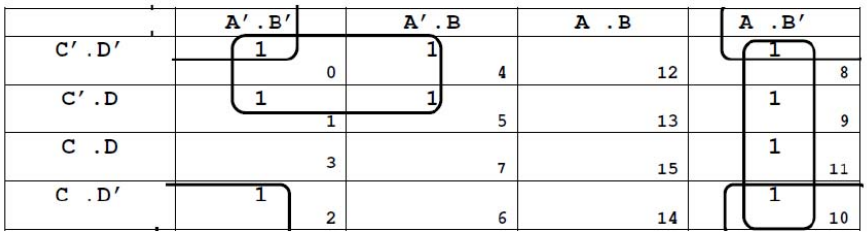
3.



The POS form is (U+Z').(V'+W').(U+V+W).(U'+W'+Z)

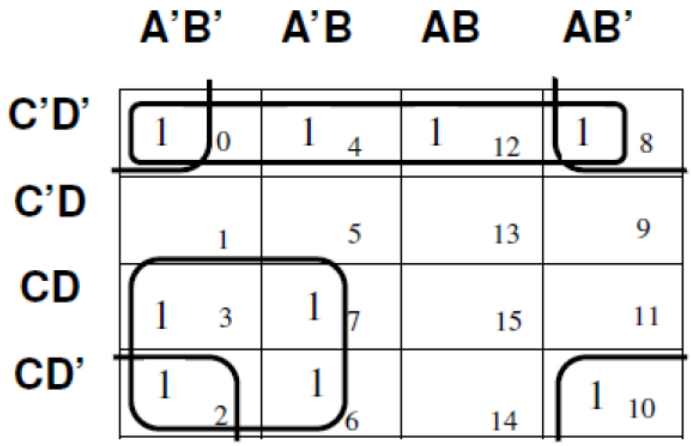
10 do yourself

4.



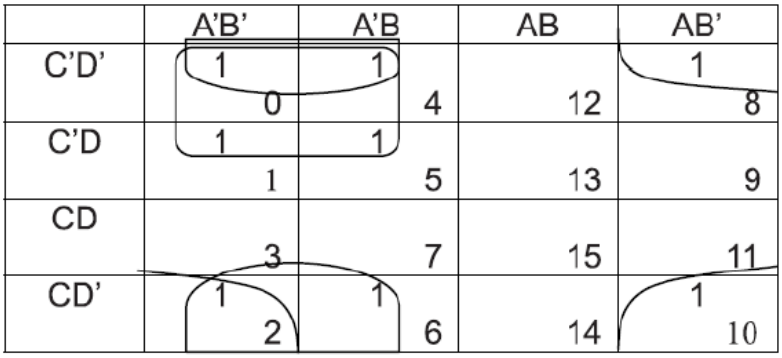
$F(A,B,C,D)= A'.C'+A.B' + B'.D'$

5.



$F= C'.D' + A'.C + B'.D'$

6.



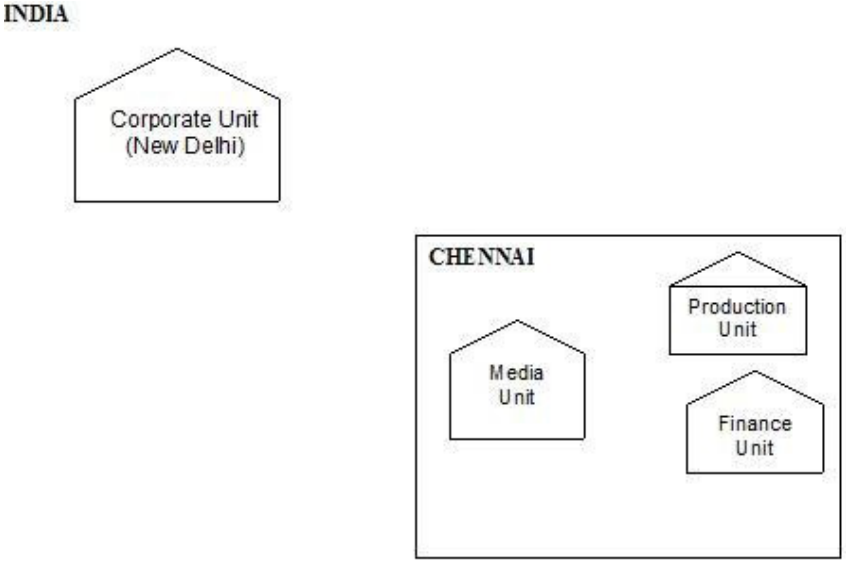
$F(A,B,C,D)=A'C'+A'D'+B'D'$

UNIT 5 : 1 Mark Questions : Communication and Network Concepts

1 Which media have the ability to communicate over oceans?
2 What is the difference between microwave and radiowave transmission?
3 Which is the special case of microwave relay system?
4 What is bleeding of signal?
5 Which communication medium is to be suggested for very effective and fast communication in guided medium?
6 In harsh industrial environment which cable would you like to use?
7 Which transmission medium is useful for sparsely populated areas?
8 In which transmission medium Bandwidth allocation is limited?
9 What is baud?
10 Which network is easy to expand?
11 In which network there is no server?
12 In Ethernet card which connection is used for coaxial cable or twisted pair or for both?
13 Which device filter the data and which device can handle different protocol?
14 Which device do not control broadcast , filter traffic and filter collision?
15. Which networking device is used for less no. of nodes or which is for more no. of nodes?

4 Marks Questions : Communication and Network Concepts

1. “China Middleton Fashion” is planning to expand their network in India, starting with two cities in India to provide infrastructure for distribution of their product. The company has planned to set up their main office units in Chennai at three locations and have named their offices as “Production Unit”, “Finance Unit” and “Media Unit”. The company has its corporate unit in New Delhi. A rough layout of the same is as follows:



Approximate distances between these Units is as follows:

From	To	Distance
Production Unit	Finance Unit	70 Mtr
Production Unit	Media Unit	15 KM
Production Unit	Corporate Unit	2112 KM
Finance Unit	Media Unit	15 KM

In continuation of the above, the company experts have planned to install the following number of computers in each of their office units:

Production Unit	150
Finance Unit	35
Media Unit	10
Corporate Unit	30

i) Suggest the kind of network required (out of LAN,MAN,WAN) for connecting each of the following office units:

- Production Unit and Media Unit
- Production Unit and Finance Unit

ii) Which one of the following devices will you suggest for connecting all the computers within each of their office units?

- Switch/Hub
- Modem
- Telephone

iii) Which of the following communication media, will you suggest to be procured by the company for connecting their local offices in Chennai for very effective (High Speed) communication?

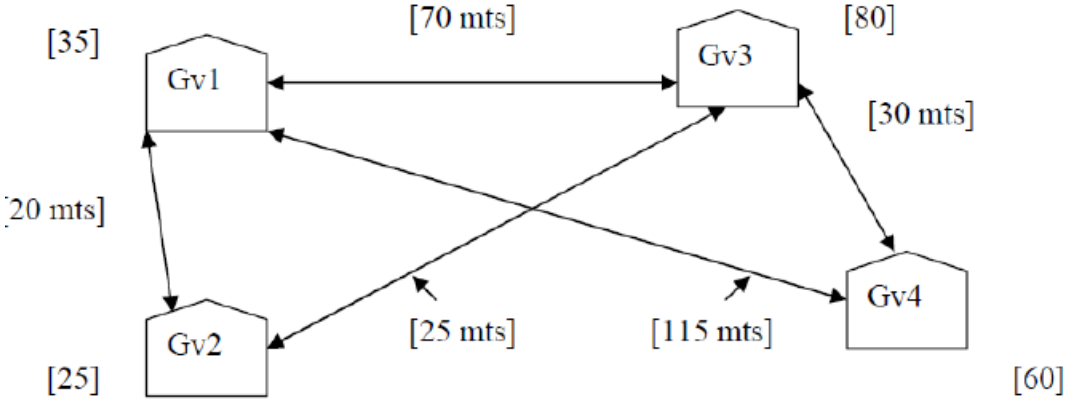
- Ethernet cable
- Optical fiber
- Telephone cable

(iv) Suggest a cable/wiring layout for connecting the company's local office units located in Chennai. Also, suggest an effective method/technology for connecting the company's office unit located in Delhi.

2. Nootan Vidya Mandir in OOTY is setting up the network between its different wings. There are 4 wings named as SENIOR(S), MIDDLE(M), JUNIOR(J) and OFFICE(O).

Distance between the various wings are given below:
Wing O to Wing S 100m
Wing O to Wing M 200m
Wing O to Wing J 400m
Wing S to Wing M 300m
Wing S to Wing J 100m
Wing J to Wing M 450m
No. of Computers
Wing O 10
Wing S 200
Wing M 100
Wing J 50
(i) Suggest a suitable Topology for networking the computer of all wings.
(ii) Name the wing where the server to be installed. Justify your answer.
(iii) Suggest the placement of Hub/Switch in the network.
(iv) Mention an economic technology to provide internet accessibility to all wings.

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



(a) Suggest a cable layout for these buildings.
(b) In each of the buildings, the management wants that each LAN segment gets a dedicated bandwidth i.e bandwidth must not be shared. How can this be achieved?
(c) The company also wants to make available shared Internet access for each of the buildings. How can this be achieved?
(d) The company wants to link its head office in GV1 building to its another office in Japan.
(i) Which type of transmission medium is appropriate for such a link?
(ii) What type of network would this connection result into?

4. Knowledge Supplement Organisation has set up its new center at Mangalore for its office and web based activities. It has 4 blocks of buildings as shown in the diagram below:

Center to center distances between various blocks

Black A to Block B 50 m
Block B to Block C 150 m
Block C to Block D 25 m
Block A to Block D 170 m
Block B to Block D 125 m
Block A to Block C 90 m
Number of Computers
Black A 25
Block B 50
Block C 125
Block D 10

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

e2) Suggest the most suitable place (i.e. block) to house the server of this organisation with a suitable reason.

e3) Suggest the placement of the following devices with justification

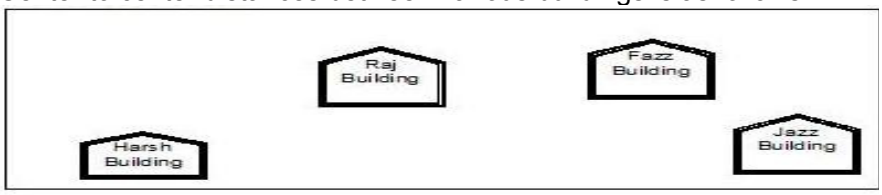
(i) Repeater

(ii) Hub/Switch

e4) The organization is planning to link its front office situated in the city in a hilly region where cable connection is not feasible, suggest an economic way to connect it with reasonably high speed?

5. Ravya Industries has set up its new center at Kaka Nagar for its office and web based activities. The company compound has 4 buildings as shown in the diagram below:

Center to center distances between various buildings is as follows:



Harsh Building to Raj Building 50 m
Raz Building to Fazz Building 60 m
Fazz Building to Jazz Building 25 m
Jazz Building to Harsh Building 170 m
Harsh Building to Fazz Building 125 m
Raj Building to Jazz Building 90 m
Number of Computers in each of the buildings is follows:
Harsh Building 15
Raj Building 150
Fazz Building 15
Jazz Bulding 25

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

e2) Suggest the most suitable place (i.e. building) to house the server of this organisation with a suitable reason.

e3) Suggest the placement of the following devices with justification:

(i) Internet Connecting Device/Modem

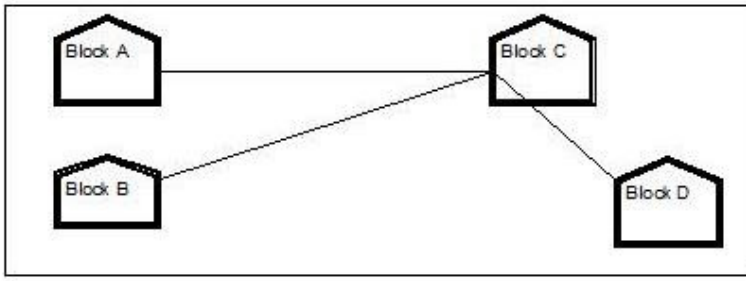
(ii) Switch

e4) The organisation is planning to link its sale counter situated in various parts of the same city, which type of network out of LAN, MAN or WAN will be formed? Justify your answer.

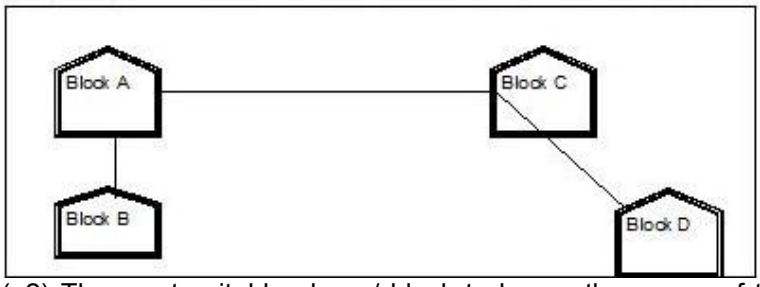
Answers of 1 mark Questions: LEFT FOR THE STUDENTS TO TRY THEMSELVES TO REACHE TO ANSWERS

Answers: 4 Marks Communication and Network Concepts

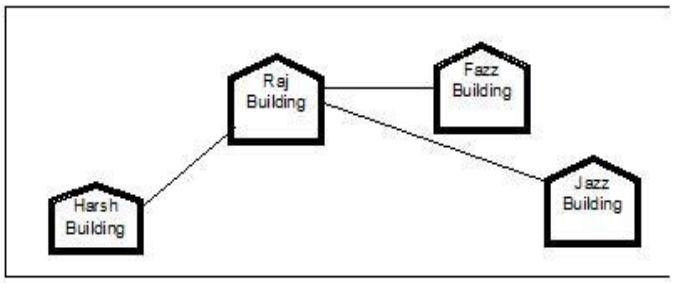
1. (i)(a) Production Unit and Media Unit :MAN
(b)Production Unit and Finance Unit:LAN
(ii) Switch/Hub
(iii) Optical fiber
(iv) Optical Fiber/Star Topology Wireless/Satellite Link/Leased Line
2. (i) Star or Bus or any other valid topology.
(ii) Wing S, because maximum number of computers are located at Wing S.
(iii) Hub/ Switch in all the wings.
(iv) Coaxial cable/Modem/LAN/TCP-IP/Dialup/DSL/Leased Lines or any other valid technology.
3. (a) Total cable length required for this layout = 75 mts
(b) To give dedicated bandwidth, the computers in each building should be connected via switches as switches offer dedicated bandwidths.
(c) By installing routers in each building, shared internet access can be made possible
(d) (i) Satellite as it can connect offices across globe.
(ii) WAN (Wide Area Network)
4 (e1) (Any of the following option)
Layout Option 1



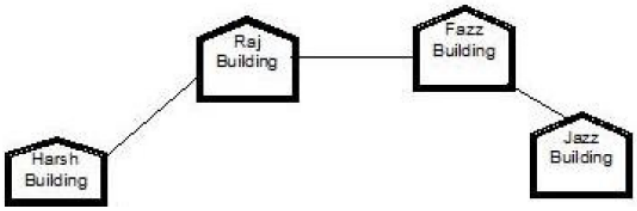
Layout Option 2



(e2) The most suitable place / block to house the server of this organisation would be Block C, as this block contains the maximum number of computers, thus decreasing the cabling cost for most of the computers as well as increasing the efficiency of the maximum computers in the network.
(e3) (i) For Layout 1, since the cabling distance between Blocks A and C, and that between B and C are quite large, so a repeater each, would ideally be needed along their path to avoid loss of signals during the course of data flow in these routes. For layout 2, since the distance between Blocks A and C is large so a repeater would ideally be placed in between this path
(ii) In both the layouts, a hub/switch each would be needed in all the blocks, to
interconnect the group of cables from the different computers in each block.
(e4) The most economic way to connect it with a reasonable high speed would be to use radio wave transmission, as they are easy to install, can travel long distances, and penetrate buildings easily, so they are widely used for communication, both indoors and outdoors. Radio waves also have the advantage of being omni directional, which is they can travel in all the directions from the source, so that the transmitter and receiver do not have to be carefully aligned physically.
5. (e1) Any one layout
Layout option1



Layout option 2



(e2) The most suitable place / block to house the server of this organisation would be Raj Building, as this block contains the maximum number of computers, thus decreasing the cabling cost for most of the computers as well as increasing the efficiency of the maximum computers in the network.

(e3)(i) Raj Building

(ii) In both the layouts, a hub/switch each would be needed in all the buildings, to interconnect the group of cables from the different computers in each block

e4) MAN, because MAN (Metropolitan Area Networks) are the networks that link computer facilities within a city

UNSOLVED QUESTIONS

INDEX

UNIT 1 : 1 MARK QUESTIONS : PROGRAMMING IN C++

1. 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();
}
```

2. Observe the program segment given below carefully and fill the blanks marked as Statement1 and Statement2 using seekp( ) and seekg( ) functions for performing the required task.

```
#include <fstream.h>
    class Item
    {
        int Imno; char Item[20];
        public:
        //Function to search and display the content from a particular record number
        void Search (int) ;
        //Function to modify the content of a particular record number
        void Modify(int);
    };
void Item :: Search (int RecNo)
    {
        fstream File;
        File.Open("STOCK.DAT" , ios :: binary | ios :: in);
        File.seekg(0,ios::beg);
        ------------------------------------        //statement 1
        Cout <<Ino <<" = = >" << Item << endl;
        File.close ( );
    }
void Item :: Modify (int RecNo)
    {
        fstream File;
        File.open ("STOCK.DAT", ios ::binary | ios :: in | ios :: out);
        cin>> Ino;
        cin.getline(Itm,20 );
        File.seekp(RecNo);
        ------------------------------------        //Statement 2

        File.close ( );                                                    }
```

3. Observe the following program segment given below carefully fill the blanks marked as Statement1 and Statement2 using seekg()and tellg( )functions for performing the requested task:

```
class Railway
{
```

```
        int Tno,
        char Tname[20];
  public:
        // Function1 to count total number of records.
        int CountRec( );
  };
  int Railway :: CountRec( )
  {
        fstream File;
        File.open("Rail.Dat", ios::in | ios::binary);
        ……..// Statement 1
        int bytes =……..// Statement 2
        int count = bytes / sizeof(Item);
        File.close( );
          return count;
   }
```

4. In the following C++ program, what will be the maximum and minimum value of r generated with the help of random function?

```
#include<iostream.h>
#include<stdlib.h>
void main()
{
        int r;
        randomize();
        r=random(20)+random(2);
        cout<<r;
}
```

5. Observe the program segment given below carefully, and answer the question that follows :

```
class Member
{       int Member_no;
        char Member_name[20];
        public :
        void enterdetails{) ;
        void showdetails();
        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.

6. Observe the program segment given below carefully , and answer the question that follows(1)

```
class Member
{     int Member_no ;  char Member_name[20] ;
public :
       void enterdetails ( ) ;        //function to enter Member details
       void showdetails ( ) ;       //function to display Member details
      int RMember_no( )         //function to return Member_no
      {return Member_no ;}} ;
void Update (Member NEW)
{     fstream File ;
       File.open("MEMBER.DAT" , ios :: binary I ios :: in I 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( ))
                 {  _____________  // Statement 1
                     _____________  // Statement 2
                     Found = 1 ;
                }
                else
                     File.write((char*) & OM, sizeof(OM)) ;
    }
    if (!Found)
                  cout<<"Record for modification does not exist" ;
    File.close( ) ;
}
```

2 MARKS QUESTIONS : PROGRAMMING IN C++

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

```
#include<iostream.h>
void main( )
{
struct movie
{ char movie_name[20];
char movie_type;
int tickets=100;
} MOVIE;
gets(movie_name);
gets(movie_type);
}
```

2. Find the output of the following program

```
#include<iostream.h>
void Modify(int &a, int b=10)
```

```
{ if(b%10==0)
a+=5;
for(int i=5;i<=a;i++)
cout<<b++<<.:.;
cout<<endl;
 }
void Disp(int x)
{
 if(x%3==0)
Modify(x);
else
Modify(x,3);
}
void main()
{ Disp(3);
Disp(4);
Modify(2,20);
}
```

3. Rewrite the following program after removing the syntax error(s), if any. Underline each correction

```
#include<iostream.h>
int main()
{
        struct student
        {
          int. rno, mark;
          }stu;
 student stuA= (1001,49);
        student stuB= stuA;
          if (stuA!= stuB)
          stuA.mark= 55;
           else
          stuB.mark= 25;
      cout<<stuA.mark<<stub.mark;
 }
```

4. Find the output of the following program:

```
#include<iostream.h>
 void main()
 {
    int list[5];
*list=5;
 for(int i=1; i<5;i++)
*(list+i)= *(list+i-1)*i;
 cout<<"\n";
 for(i=0;i<5;i++)
 cout<<""<<*(list+i);
 }
```

5. Find the output of the following program

```
#include<iostream.h>
void change (int x[4],int i)
 {
    x[i]=x[i]*i;
  }

   int main()
   {
     int a[]={1,11,111,1111};
```

```
    for(int k=0;k<4;k++)
    {
       change(a,k);
        cout<<a[k]<<endl;
     }
   }
```

6. Answer the questions (i) and (ii) after going through the following program:

```
#include <iostream.h>
#include<stdio.h>
 #include<string.h>
 class shares
 { char company[20];
     int no_of_shares;
  public:
  shares()                                  / / function 1
  {
     strcpy(company,"      ");
      no_of_shares = 0;
    }

   shares(char factory[], int n)            / / function 2
    {
       strcpy(company,factory);
       no_of_shares = n;
     }

    void read_data()                        / /  function 3
    {
      gets(company);
       cin>>no_of_shares;
     }

     ~shares()                              / /  function 4
     {
       cout<<"Share market holiday"<<endl;
      }
};
```

(i) Write the statements to call function 1 and function 2.

(ii) What is the role of function 3 and function 4 in this class shares?

7. Rewrite the following program after removing the syntactical errors (if any).Underline each correction.

```
#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();
        }
```

8. Find the output of the following program:

```
#include<iostream.h>
void main()
{
int Numbers[] = {2,4,8,10};
int *ptr = Numbers;
for (int C = 0; C<3; C++)
{
cout<< *(ptr+1)-1 << "@";
ptr++;
}
cout<<endl;
for(C = 0; C<4; C++)
{
(*ptr)=(*ptr * 2);
--ptr;
}
for(C = 0; C<4; C++)
cout<< Numbers [C]<< "#";
cout<<endl;
}
```

9. Observe the following program RANDNUM.CPP carefully. If the value of VAL entered by the user is 10, choose the correct possible output(s) from the options from i) to iv) and justify your option.

```
//program RANDNUM.CPP
#include<iostream.h>
#include<stdlib.h>
#include<time.h>
void main()
{
        randomize();
        int VAL, Rnd; int n=random(2);
        Rnd=8 + random(sizeof(VAL)) * 1;
        while(n<Rnd)
        {
                cout<<n<< "\t";
                n++;
        }
}
```

output options:

i) 1 2 3 4 5 6 7 8 9 10 11 12 13
ii) 0 1 2 3 4 5 6 7 8
iii) 1 2 3 4 5
iv) 1 2 3 4 5 6 7 8

10. Answer the questions (i) and (ii) after going through the following program:

```
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.

(ii) How would Function 1 and Function 2 get executed ?

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

```
$include<iostream.h>
class ERROR
{
        char E_Name[20];
        int E_No=10;
public:
        ERROR( ) { };
        void show_ERROR( )
        {
                gets(E_Name); cin>>E_No;
        }
};
void main( );
{
        ERROR *E;
        E.show( );
}
```

12. Find the output of the following program:

```
#include<iostream.h>
#include<ctype.h>
void main( )
{
        char *s = "Polymorphism";
        int L = strlen ( s );
        for ( int C = 0 ; C < L ; C ++)
        {
                if (islower ( s [C] )
                        s[C] = toupper( s[C]);
                else if (C % 2 == 0)
                        s[C] = 'E';
                else
                        s[C] = tolower( s[C]);
}
```

13. Read the program carefully and select the possible output(s) from it:

```
#include<iostream.h>
#include<stdlib.h>
const int MAX = 3;
void main( )
{
        randomize ( );
         int digit;
        digit =80 + random (MAX );
        for ( int R = digit ; R >= 80 ; R --)
                cout << R << "$" ;
        cout<<endl;
```

```
}
```
(i) 83$82$81$80$
(ii) 80$81$82$
(iii) 80$81$
(iv) 81$80$

14. Answer the questions (i) and (ii) after going through the following class:

```
class Bag
{   int pockets;
public:
        Bag ( )                                          // Function 1
        {
                pockets = 30;
                cout << " The Bag has pockets"<<endl;
        void Company ( )                                 // Function 2
        {
                cout << " The company of the Bag is VIP "<< endl;
                }
        Bag (int D)                                      // Function 3
        {
                pockets = D;
                cout<<" The Bag has pockets"<<endl;
        }
        ~ Bag ( )                                        // function 4
        {
                cout <<" Thanks!"<< endl;
        }
};
```

(I)In object Oriented Programming, what is function 4 referred as and when does it get invoked / called?
(II)Which concept is illustrated by Function 1 and function 3 together? Write an example illustrating the call of these functions

15. Write a function in C++ to count and displays the articles 'a', 'an' and 'the' present in a text file ARTICLE.TXT.For Example:
If the file contains:"He is Ram. He is a good boy. He studies in the class 12^{th}. He has an elephant. The elephant is very strong animal."
Then the output should be:
Article 'a' = 1
Article 'an'=1
Article 'the'=2

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

```
#include<iostream.h>
#define SIZE =10
VOID MAIN()
{
        int a[SIZE]={10,20,30,40,50};
        float x=2;
        SIZE=5;
        for(int i=0;i<SIZE;i++)
                cout<<a[i]%x;
}
```

17. Find the output of the following programs :

i)
```
#include<iostream.h>
#include<string.h>
struct Student
```

```
{
        int rno;
        char name[20];
};
void main()
{
        student a[2]={1,"Amit",}{2,"Sumit"}};
        for(int i=0;i<2;i++)
        {
                cout<<"\n Rno"<<a[i].rno;
                cout<<"\n Name ";
                for(int j=0;j<strlen(a[i].name);j++)
                        cout<<a[i].name[i]<<" ";
        }
}
```

ii)

```
#include<iostream.h>
int a=10;
void main()
{
        void demo(int &,int,int*);
        int a=20,b=5;
        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;
}
```

18. 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 1.

(ii) Write complete definition for constructor 2.

19. Identify the errors in the following program.

```
#include<iostream.h>
void main()
{       int n = 44;
        int *ptr = &n;
        ++(*ptr);
        int  *const cptr = &n;
        ++(*cptr);
        ++cptr;
        const int kn=88;
        const int *ptrc = &kn;
        ++(*ptrc);
        ++ptrc;
```

```
        const int *const cptrc =&kn;
        ++(*cptrc);
        ++cptrc;
    }
```

20. Give the output of the following program segment (Assume all required header files are included in the program)

```
    void main()
    {   char *name,*name1;
        int l=0;
        name="Windows98";
        l = strlen(name);
        cout<<endl;
        for (int asc=90;asc>=65;asc--)
        {   for(int i=0;i<l;i++)
        {
if (name[i]==char(asc) || (name[i]==char(asc+32)))
cout<<name[i];
  }  }
  cout<<endl;       }
```

21. Write the output of the following program:

```
#include<iostream.h>
int func (int &x, int y=10)
{
if(x%y==0) return ++x; else return y--;
}
void main()
    { int p=20,q=23;
    q=func(p,q);
    cout<<p<<" "<<q<<endl;
    p=func(q);
    cout<<p<<" "<<q<<endl;
    q=func(p);
    cout<<p<<" "<<q<<endl;
    }
```

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

```
#include<conio.h>
  void main( );
  {  structure movie
     {   char movie_name[25];
         char movie_type;
   int ticket_cost=50;
}M
gets(movie_name);
gets(movie_type);    }
```

23. Find the output of the following program.

```
#include<iostream.h>
  void Withdef(int HisNum=29)
  {  for(int I=12;I<=HisNum;I+=7)
       cout<<I<<",";
      cout<<endl;
  }
```

```
void Control(int &MyNum)
{   MyNum+=8;
    Withdef(MyNum);      }
void main( )
{
int YourNum=16;
Control(YourNum);
Withdef( );
cout<<"Number="<<YourNum<<endl;     }
```

24. Observe the following program TEST.CPP carefully, if the value of Disp entered by the user is 22, choose the correct possible output(s) from the options from (i) to (iv), and justify your option.

```
//program: TEST.CPP
    #include<stdlib.h>
    #include<iostream.h>
    void main( )
    {  randomize( );
       int Disp,Rnd;
       Cin>>Disp;
       Rnd=random(Disp)+15;
       for(int N=1,i=3;i<=Rnd; i+=4,N++)
        cout<<N<<"   ";     }
```

Output Options:

(i) 1 (ii) 1 2 3 4

(iii) 1 2 (iv) 1 2 3

25. Find the output of the following program:

```
#include<iostream.h>
#include<string.h>
class State
{
    char *state_name;
    int size;
  public:
    State( ){size=0;state_name=new char [size+1];}
    State(char *s)
    {  size=strlen(s);
       state_name=new char[size+1];
       strcpy(state_name,s);
     }
     void display( )
     { cout<<state_name<<endl;
      }
      void Replace(state &a, state &b)
      { size=a.size+b.size;
        delete state_name;
        state_name=new char[size+1];
        strcpy(state_name, a.state_name);
        strcat(state_name, b.state_name);
       }
};

void main( )
{
  char *temp="Calcutta";
  State state1(temp),state2("Hyderabad"),state3("Chennai"),S1,S2;
  S1.Replace(state1,state2);
```

```
    S2.Replace(S1,State3);
    S1.display( );
    S2.display ( );
    S2.Replace(state2,state3);
    S2.display( );
}
```

26. Answer the questions (i) and (ii) after going through the following program:

```
class Date
{   int day,month,year;
 public:
     Date(Date &D);                    //Constructor 1
     Date(int a,int b,int c)           //Constructor 2
     {  day=a;
        month=b;
        year = c;
      }
 };
```

(i) Write complete definition for Constructor 1

(ii) Create an object, such that it invokes Constructor 2

3 MARKS QUESTIONS : PROGRAMMING IN C++

1.Find the output of the following program (Assuming that all required header files are included)

```
void main( )
{
char * NAME = “admiNStrAtiOn”;
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] = NAME[x -1];
else
NAME[x]--;
cout<<NAME <<endl;
}
```

2. Find the output of the following program:

```
#include<iostream.h>
#include<string.h>
class country
{       char *country_name;
        int length;
        public:.
        country ( ) {length =0; country_name=new char [length+1];}
        country (char *s)
        {       length = strlen(s); country_name=new char [length +1];
                strcpy (country_name, s);
        }
        void display () { cout<< country_name <<endl;}
        void Replace (country & a, country & b)
        {       length a.length + b.length;
                delete country_name;
                country_name=new char [length + 1];
                strcpy (country_ name, a.country_name);
```

```
            strcat (country_name, b.country name);
            }
};
void main ( )
{     char * temp = “India”;
      country country1 (temp), country2 (“Nepal”), country3 (“China”), S1,S2;
      S1.Replace (country1, country2);
      S2.Replace (S1,country3);
      S1.display( );
      S2.display ( );
}
```

3. Find the output of the following program:

```
void main( )
{
        char *poet= "SakESpHerE" ;
        for (int i=0;i<strlen(poet);i++)
        {
                if(islower(poet[i]))
                poet[i]=poet[i-1];
                else if( isupper(poet[i]))
                if(poet[i]= ='S')
                        poet[i]='X';
                else if(poet[i]= ='E')
                        poet[i]=toupper(poet[i-1]);
                else
                        poet[i]--;
        }
        cout<<poet;  }
```

4. Write a function in C++ to search for BookNo from a binary file “BOOK.DAT”, assuming the binary file is contained the objects of the following class:

```
class BOOK
{
        int Bno; char Title [20];
        public :
        int Rbno ( ) { return Bno; }
        void Enter ( ) { cin >> Bno; gets (Title); }
        void Display ( ) { cout << Bno <<Title <<endl; }
};
```

5. Find the output of the following program:

```
        3
#include<iostream.h>
int main( )
{
        void NewValue(int, int *);
        int val[ ] = {3, 5, 7, 9, 11};
        int i, num=5;
        for( i = 0 ; i < 5 ; i++)
        {
                NewValue(val[i], &num);
                cout<<val[i] << “\t” << num << “\n”;
        }
}
void NewValue( int A, int *B)
{
        A = *(B) += 3;
```

}

6. Assuming the given definition of class HOTELDATA, write functions in C++ to perform the following:

```
class HOTELDATA
{               int room;
                char name[20];
                int duration;
        public:
                void checkins();
                void display(); };
```

Checkins( ) function to allow user to enter the data of customers (objects of class HOTELDATA) and write them to a binary file "HOTEL" and display( ) function allows us to read from the binary file and display on the screen.

4 MARKS QUESTIONS : PROGRAMMING IN C++

1. Define a class **Travel** in C++ with the description given below:
Private members:
plancode of type long
place of type characters array
number_of_travellers of type integer
number_of_buses of type integer
Public members:
A constructor to assign initial values of plancode as 1001, place as “Kolkata”, number_of_travellers as 5 and number_of_buses as 1
A function newplan( ) which allows user to enter plancode , place and number_of_travellers and also assign the number_of_buses as per the following conditions:

number_of_travellers	**number_of_buses**
less than 20	2
equal to and more than 20 and less than 40	3
equal to and more than 40	4

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

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

```
4
class Goods
{ int id;
protected :
char name[20];
long qty;
void Incr(int n);
public :
Goods();
~Goods();
void get(); };
class Food_products : public Goods
{ char exp_dt[10];
protected :
int id;
int qty;
public :
void getd();
void showd(); };
class Cosmetics : private Goods
{ int qty;
char exp_date[10];
protected :
int id;
public :
~Cosmetics();
Cosmetics();
void show();
};
```

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

3. Define a class named DRAMA in C++ with the following description
   Private members
   SHOW_NO integer
   NAME_OF_THE_DRAMA Array of characters (String)
   DAY integer (Total number of days the same drama is shown)
   DAILY_COLLECTION float
   TOTAL_COLLECTION float
   Public Members
   input_data : A user defined function to read an object of ENTRY type
   print_data : A function to display the details of an object
   update_data: A function to update the total collection and daily collection once the day changes. Total collection will be incremented by daily collection and daily collection is made Zero
4. Answer the questions (i) to (iv) based on the following code:

```
          (4)
  class engineering
  {
  private:
          char streamcode[5];
  protected:
          int seats;
          void allot();
  public:
           engineering();
            void streamread();
             void streamwrite();
 };
 class dept : protected engineering
 {
       char deptname[20];
       int strength;
  public:
          dept();
          void deptread();
          void deptwrite();
  };
  class course: public dept
  {
          char coursename[20];
          float fees;
  public:
          course();
          void courseread();
          void coursewrite();
  };
```

   (i) Which type of inheritance is shown in the above example?
   (ii) How many bytes will be required by an object of the class dept and course?
   (iii) Write the name of all the data members accessible from member functions of the class course.
   (iv) Write the members which are accessible from the object of the class dept.
5. Define a class **Departmental** with the following specification :
   private data members
   Prod_name string (45 charactes) [Product name]
   Listprice long
   Dis_Price long [Discount Price]
   Net long [Net Price]
   Dis_type char(F or N) [Discount type]

Cal_price() – The store gives a 10% discount on every product it sells. However at the time of festival season the store gives 7% festival discount after 10% regular discount. The discount type can be checked by tracking the discount type. Where 'F' means festival and 'N' means Non-festival .The Cal_price() will calculate the Discount Price and Net Price on the basis of the following table.

Product Name	List Price(Rs.)
Washing Machine	12000
Colour Television	17000
Refrigerator	18000
OTG	8000
CD Player	4500

public members

- ➔ Constructor to initialize the string elements with "NULL", numeric elements with 0 and character elements with 'N'
- ➔ Accept() - Ask the store manager to enter Product name, list Price and discount type . The function will invoke Cal_price() to calculate Discount Price and Net Price .
- ➔ ShowBill() - To generate the bill to the customer with all the details of his/her purchase along with the bill amount including discount price and net price.

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

```
class X
{
     int P[20];
     float b;
    protected :
       char str[20];
    public:
      X();
      void Read();
      void  Display();
};
class  E: protected X
{
     int No;
     protected :
       float S;
     public :
       E();
       void EDetails();
       void DDetails();
 };
class P : public  X
{
      int Pno;
      long  double b2;
     pubic :
         P();
          void Accept();
          void Show();
};
void main()
 {
```

P obj;
}

i) Mention the members accessible by the obj declared in main()

ii) What is the size of obj in bytes? What type of inheritance is depicted in the given snippet?

iii) If E if inherited in private visibility mode and P in public visibility mode from E. name the members that can be accessed by the obj.

iv) Name the member functions that can access the data members str, if the inheritance is in multi-level format

7. Define a class Computer in C++ with following description: private Members:

- Processor _speed
- Price
- Processor_type

Public Members:

- A constructor to initialize the data members.
- A function cpu_input() to enter value of processor_speed.
- A function void setcostANDtype( ) to change the speed of the processor and also find the cost and type depending on the speed:

Processor_speed	Price	Processor_type
4000 MHz	Rs 30000	C2D
<4000 & >=2000	Rs 25000	PIV
< 2000	Rs 20000	Celeron

- A function cpu_output() to display values of all the data members.

8. Answer the questions (i) to (iv) after going through the following class:

```
                4
class Sports
{
        char Category[10];
        char Date_of_Activity[10];
        char Name[20];
public:
        Sports( );
        void EnterDetails( );
        void ShowDetails ( );
};
class MohanClub : public Sports
{
protected:
        char Player_Name[30];
        char Player_Address[20];
public:
        float fees;
        MohanClub( );
        void EnterClubDetails( );
        void ShowClubDetails( );
};
class HealthClub : public MohanClub
{
        int Facility;
        char F_Name[20];
```

```
public:
        HealthClub( );
        void EnterHClubDetails( );
        void showHClubDetails( );
};
```

(i) How many bytes will be required by an object of class HealthClub and an object of class MohanClub respectively?

(ii) Write names of all the data members which are accessible from the object of class HealthClub?

(iii) Write the name of ALL the members accessible from the member function of class MohanClub.

(iv) Write the name of all the member functions which are accessible from the object of class HealthClub?

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

Data Members:

Event_no	integer
Description	char(30)
Score	integer
qualified	char

Member functions:

- A constructor to assign initial values Event_No number as 101, Description as “State level”, Score is 50 and qualified as ‘N’.
- Input(), To take the input for event_no, description and score.
- Award(int), To award qualified as ‘Y’, if score is more than the cutoffscore passed as argument to the function else ‘N’.
- Show(), To display all the details.

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

```
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, which are accessible by the objects 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) Is the constructors of class Employee will copied in class Payment? Due to inheritance.

11. Define a class BALANCED_MEAL in C++ with following description:

Private Members:

Access number	Integer
Name of Food	String of 25 characters
Calories	Integer
Food type	String
Cost	Float
AssignAccess( )	Generates random numbers between 0 to 99 and return it.

Public Members

- A function INTAKE( ) to allow the user to enter the values of Name of Food, Calories, Food type cost and call function AssignAccess() to assign Access number.
- A function OUTPUT( ) to allow user to view the content of all the data members, if the Food type is fruit.

12. Consider the following declarations and answer the questions given below: **4**

```
class Mydata
{       protected:
        int data;
        public:
        void Get_mydata(int);
        void Manip_mydata(int);
        void Show_mydata(int);
        Mydata( );
        ~Mydata( );                     };
class Personal_data
{
        protected:
        int data1;
        public:
        void Get_personaldata(int);
        void Show_personaldata(int);
        Mydata1( );
        ~Mydata1( );                    };
class Person: public Mydata, Personal_data
{
        public:
        void Show_person(void);
        person( );
        ~person( );                     };
```

i) How many bytes will be required by an object belonging to class Person?

ii) Which type of inheritance is depicted in the above example?

iii) List the data members that can be accessed by the member function Show_person( )

iv) What is the order of constructor execution at the time of creating an object of class Person?

13. Define a class PhoneBill in C++ with the following descriptions.

Private members:

CustomerName of type character array
PhoneNumber of type long
No_of_units of type int
Rent of type int
Amount of type float.

calculate( ) This member function should calculate the value of amount as Rent+ cost for the units.

Where cost for the units can be calculated according to the following conditions.

No_of_units	Cost
First 50 calls	Free
Next 100 calls	0.80 @ unit
Next 200 calls	1.00 @ unit
Remaining calls	1.20 @ unit

Public members:

* A constructor to assign initial values of CustomerName as “Raju”, PhoneNumberas 259461, No_of_units as 50, Rent as 100, Amount as 100. * A function accept( ) which allows user to enter CustomerName, PhoneNumber,No_of_units And Rent and should call function calculate( ). * A function Display( ) to display the values of all the data members on the screen.

14. Answer the questions (i) to (iv) based on the following code: (4)

```
class Teacher       {
    char TNo[7],TName[25],Dept[12];
    int Wload;
  protected:
    double Sal;
    void AssignSal(double);
  public:
    Teacher( );
    Teacher(Double S);
    void TeaNew( );
    void TeaDisplay( );             };
class Student
{
    char ANo[6],SName[15],Group[7];
  protected:
    int Att,Total;
  public:
    Student( );
    void StuAccept( );
    void StuDisplay( );             };
class School: public Student, private Teacher
{
    char SchCode[9],SchName[15];
  public:
    School( );
    void SchAccept( );
    void SchDisplay( );         };
```

(i) How many bytes will be reserved for an object of type School?(ii) Name the members that can be called by object of type School.

(iii)Which type of inheritance is depicted by the above example?(iv) Identify the member function(s) that cannot be called directly from the objects of class School from the following: (a) TeaNew( ) (b) StuAccept( ) (c) SchDisplay( ) (d)AssignSal( )

UNIT 2 : 3 MARKS QUESTIONS : DATA STRUCTURE

1. Write a function in C++ which accepts an integer array and its size as arguments and change all the even number with twice and odd with thrice.Example: if an array of five elements initially contains the element as
2,4,1,5,7
then the function should rearrange the array as
4,8,3,15,21

2. W rite 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 integer in the following format:

3

if the array is 1,2,3,4,5,6
the resultant 2D array is given below
1 2 3 4 5 6
1 2 3 4 5 0
1 2 3 4 0 0
1 2 3 0 0 0
1 2 0 0 0 0
1 0 0 0 0 0

if the array is 1,2,3
the resultant 2D array is given below
1 2 3
1 2 0
1 0 0

3. Write a function in C++ which accepts an integer array and its size as arguments/ parameters and then assigns the elements into a two dimensional array of integers in the following format:

(3)

If the array is 1, 2, 3, 4, 5, 6
The resultant 2 D array is given below

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

If the array is 1, 2, 3
The resultant 2 D array is given below

0	0	1
0	2	1
3	2	1

5. Write a C++ function RevDup(int [], int) to remove the duplicate occurrence of the value Present in an integer array, passed to the function as a parameter.
For e.g. If array initially is X[] = { 1,1,1,7,5,2,2,6}
After removing duplicate values the array will be X {} = {1, 7, 5, 2, 6}

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

```
struct Employee
{
        int ENo;        char Ename[25];        float Salary;
};
```

7. Write a function in C++ which accepts a character array and its size as arguments and reverse that array without using second array and library function.
Example : if the array is having: “Computer Science”
Then after reversal it should rearranged as: “ecneicS retupmoC”

8. WAP that accept an array of 10 integers with size. The function fins a particular number from the array by using the binary search method

9. Write a function in C++ which accepts an integer array and its size as arguments/parameters and assigns the elements in to two dimensional array of integers in the following format: **4**
If the array is **1,2,3,4,5,6** ,then the resultant 2D array should be :

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

10. Suppose A,B,C are arrays of size m,n,m+n respectively. Array A is stored in ascending order and array B is in descending order. Write a function to receive 3 arrays and their sizes to store the elements of A and B into C in descending order.
11. Write a user defined function to sort the array (same as above) using insertion sort in descending order. Give the array status after each iteration
12. . Write a user defined function to sort the array(same as above) using bubble sort in descending order. Give array status after each iteration
13. Using a two dimensional A[n x n], write a function to prepare one dimensional arrayArray[n2] that will have all the elements of A as if they were stored in Column major order.
14. Write a function to search for a given number in a given array ARR[n] using linear search technique. If the number is found, move it at the top of the array. If the number is not found, insert it at the end of the array.
15. What would be the output of the following? Assume that the array starts at location 5700 in the memory?

```
# include<iostream.h>
void main()
{
int ab[3][4]={ 5,6,7,8,1,2,3,4,9,10,0,11};
cout<<"\n"<<*ab[0]<<" "<<*(tab[0]+1);
cout<<"\n"<<*(*(ab+0)+1);
}
```

16. Write a function that reads 10 integers into the array A. Use another integer array P of same size to store each index of the array A in the following way: The index of the first smallest element in A is stored at index 0 of P, the index of the next smallest element in A is stored at index 1 of P and so on. Print the next smallest element in A ordered in the sequence given by each succeeding index stored in P.
17. Write a user defined function to sort the given array using Selection sort mechanism. int A[ ] = {10,14,126,23,26,33,44,48,50,55,60,66};
    Print array after each iteration.
18. Write a user defined function to search for 55 and 23 in the following array.
    10,14,126,23,26,33,44,48,50,55,60,66 Make use of binary search method.

4 MARKS QUESTIONS : DATA STRUCTURE

1. An array A[40][10] is stored in the memory along the column with each element occupying 4 bytes. Find out the Base address and address of the element A[3][6] if the element A[30][10] is stored at the address 9000.

2. 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 Row wise.*
3. An array M[-3…18][-8….37] 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 M[2][5], if the element M[5][10] is stored at address 4000

4. An array P[10][10] is stored in the memory along the column with each element occupying 2 bytes of storage, find out the base address and address of the location P[5][5], if the element P[2][2] is stored at the memory location at 1000.

5. An array A[-2..8][-2..5] is stored in the memory along the column with each element occupying 4 bytes. Find out the address of the element A[3][2].

6. If an array B[11][8] is stored as column wise and B[2][2] is stored at 1024 and B[3][3] at 1084. Find out the base address, size of an element and address of B[5][3].

7. 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.

2 Mark Questions : Linked List, Stack, Queue

1. Convert the following infix expressions to postfix expressions
    1. A + (B * C) ^ D – (E / F – G)
    2. A * B / C * D ^ E * G / H
    3. ((A*B)-((C_D)*E/F)*G
    4. A+B/(P+Q)^C/D-E/F
    5. A+B/C*D+F*G
    6. A+B-A/(B*(A-B-A*D)^B)
    7. (B+(C+D)*(E+F)/G)/H
    8. A*(B/C)/D-E-(F+G/H)
    9. (TRUE || FALSE) && ! (FALSE || TRUE)
    10. (A / B + C) / D + E / (F + G * H / I)
    11. A OR NOT B AND C
    12. (((ax/by –a/b)-dx/fx)+a)+b
    13. (((b-(c*d-e)+f))/g)+(h*j+x)
    14. A+(((B*C)*(D+E)+F*G)^(H-J)
    15. (A-B) *(C/(D-E)+F-G
2. Evaluate the following postfix expression E given below, show the contents of the stack during the evaluation
    1. E= 5,9,+2,/,4,1,1,3,_,*,+
    2. E= 80,35,20,-,25,5,+,-,*
    3. E= 30,5,2,^,12,6,/,+,-
    4. E=15, 3, 2, +, /, 7, + 2, *
    5. E=25, 8, 3, - , / 6, *, 10 +
    6. E=8, 7, -, 9, 8, *, -, 2, / , 3,4, * 2, / -
    7. E= 5,20,15,-,*,25,2,*,-
    8. IF A=2,C=3,D=2,E=5,F=4,G=6 then EFG^D+AC/- +
    9. E= 10,+,15,*,25,5,/,2+
    10. E= 7,6,2,/,+,18,-
    11. E=AB – CD + E * + WHERE A = 5, B = 3, C = 5, D = 4 AND E = 2
    12. E= 7,6, +, 8, * , 2, -, 3, * , 2, 4, * , -

3/4 Mark Questions : Linked List, Stack, Queue

1. Define member function QInsert( ) to insert and QDel( ) to delete nodes of a linked list implemented class Queue having the following Definitions:

```
Struct Node
{ char name[20];
int age;
Node *Link;
};
class Queue
{ Node *Rear, *Front;
public:
Queue( ) { Rear=NULL; Front = NULL}
void QInsert( );
void QDel( );
};
```

2. Write a function in C++ to delete a node containing names of student, from a dynamically allocated stack of names implemented with the help of following structure :

```
struct student
{
char name[20];
student *next;
};
```

3. Write a user defined function in C++ to insert an element from a dynamically allocated Queue where each node contains the long integer (schoolno) as data. Assume the following definition of SCHOOL for the same.

```
  struct SCHOOL
{
        long scno;
        SCHOOL * link;
};
```

4. Given the following class:

```
char *msg[]={"OVER FLOW","UNDER FLOW"};
class stack
{
        int top;
        stk[5];
        void err_rep(int e_num)
        {cout<<msg[e_num];}        // report error message
         public:
          void init() {top=0;}
          void push(int);              // put the new value in to stk
           void pop();                 // get the top valued from stk
};
```

Define pop outside the stack. In your definition take care of underflow condition. Function pop has to invoke err_rep to report over flow.

5. Write a function to insert and delete a set of integer values in a circular queue and display them.

6. Consider the following portion of a program, which is implemented a linked list of library. Write the definition of function PUSH( ), to insert a new node in the stack with required information:

```
  struct Library
  {
 int id;   char name[20];        Library *Link;
  };
```

7. Write a function in C++ to delete a node containing names of student, from a dynamically allocated stack of names. The function receives the value of top by reference. The stack is implemented with the help of following structure :

```
struct student
{
        char name[20];
        student *next;
};
```

8. Write a function to insert a set of integer values in a circular queue and display them.
9. Give the necessary declarations for a queue containing float type numbers; write a user defined function in C++ to insert a float type number in the queue. Use linked representation of queue
10. Write a function in C++ to delete a node containing city's information (ie city name and its population) from a dynamically allocated Queue of cities.

Unit 3 : Database and SQL :

1 OR 2 marks questions

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

Database and SQL : 6 OR 8 marks questions

1.

Table : SchoolBus

Rtno	Area_overed	Capacity	Noofstudents	Distance	Transporter	Charges
1	Vasant kunj	100	120	10	Shivamtravels	100000
2	Hauz Khas	80	80	10	Anand travels	85000
3	Pitampura	60	55	30	Anand travels	60000
4	Rohini	100	90	35	Anand travels	100000
5	Yamuna Vihar	50	60	20	Bhalla Co.	55000
6	Krishna Nagar	70	80	30	Yadav Co.	80000
7	Vasundhara	100	110	20	Yadav Co.	100000
8	Paschim Vihar	40	40	20	Speed travels	55000
9	Saket	120	120	10	Speed travels	100000
10	Jank Puri	100	100	20	Kisan Tours	95000

(b) To show all information of students where capacity is more than the no of student in order of rtno.
(c) To show area_covered for buses covering more than 20 km., but charges less then 80000.
(d) To show transporter wise total no. of students traveling.
(e) To show rtno, area_covered and average cost per student for all routes where average cost per student is - charges/noofstudents.
(f) Add a new record with following data:
(11, " Moti bagh",35,32,10," kisan tours ", 35000)
(g) Give the output considering the original relation as given:
(i) select sum(distance) from schoolbus where transporter= " Yadav travels";
(ii) select min(noofstudents) from schoolbus;
(iii) select avg(charges) from schoolbus where transporter= " Anand travels";
(iv) select distinct transporter from schoolbus;

2.

TABLE : GRADUATE

S.NO	NAME	STIPEND	SUBJECT	AVERAGE	DIV.
1	KARAN	400	PHYSICS	68	I
2	DIWAKAR	450	COMP. Sc.	68	I
3	DIVYA	300	CHEMISTRY	62	I
4	REKHA	350	PHYSICS	63	I
5	ARJUN	500	MATHS	70	I
6	SABINA	400	CEHMISTRY	55	II
7	JOHN	250	PHYSICS	64	I
8	ROBERT	450	MATHS	68	I
9	RUBINA	500	COMP. Sc.	62	I
10	VIKAS	400	MATHS	57	II

(a) List the names of those students who have obtained DIV 1 sorted by NAME.
(b) Display a report, listing NAME, STIPEND, SUBJECT and amount of stipend received in a year assuming that the STIPEND is paid every month.
(c) To count the number of students who are either PHYSICS or COMPUTER SC graduates.
(d) To insert a new row in the GRADUATE table: 11,"KAJOL", 300, "computer sc", 75, 1
(e) Give the output of following sql statement based on table GRADUATE:
(i) Select MIN(AVERAGE) from GRADUATE where SUBJECT="PHYSICS";
(ii) Select SUM(STIPEND) from GRADUATE WHERE div=2;
(iii) Select AVG(STIPEND) from GRADUATE where AVERAGE>=65;
(iv) Select COUNT(distinct SUBDJECT) from GRADUATE;
(f) Assume that there is one more table GUIDE in the database as shown below:

Table: GUIDE

MAINAREA	ADVISOR
PHYSICS	VINOD
COMPUTER SC	ALOK
CHEMISTRY	RAJAN
MATHEMATICS	MAHESH

g) What will be the output of the following query: SELECT NAME, ADVISOR FROM GRADUATE,GUIDE WHERE SUBJECT= MAINAREA;

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

Table: SPORTS

Student NO	Class	Name	Game1	Grade	Game2	Grade2
10	7	Sammer	Cricket	B	Swimming	A
11	8	Sujit	Tennis	A	Skating	C
12	7	Kamal	Swimming	B	Football	B
13	7	Venna	Tennis	C	Tennis	A
14	9	Archana	Basketball	A	Cricket	A
15	10	Arpit	Cricket	A	Atheletics	C

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

4.

Employees

Empid	Firstname	Lastname	Address	City
010	Ravi	Kumar	Raj nagar	GZB
105	Harry	Waltor	Gandhi nagar	GZB
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	Ritu	Tondon	Shastri Nagar	GZB
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

Write the SQL commands for the following :

(i) To show firstname,lastname,address and city of all employees living in paris

(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 Employee and empsalary , where total salary is calculated as salary+benefits.

(iv) To display the maximum salary among managers and clerks from the table Empsalary.

Give the Output of following SQL commands:

(i) Select firstname,salary from employees ,empsalary where designation = 'Salesman' and Employees.empid=Empsalary.empid;
(ii) Select count(distinct designation) from empsalary;
(iii) Select designation, sum(salary) from empsalary group by designation having count(*) >2;
(iv) Select sum(benefits) from empsalary where designation ='Clerk';

4. Write the SQL commands for the i) to iv) and write the output of the (v) on the basis of table TEACHER.

5.

Table:TEACHER

No.	Name	Age	Department	Dateofadm	Salary	Sex
1	Jugal	34	Computer	10/01/97	12000	M
2	Sharmila	31	History	24/03/98	20000	F
3	Sandeep	32	Maths	12/12/96	30000	M
4	Sangeeta	35	History	01/07/99	40000	F
5	Rakesh	42	Maths	05/09/97	25000	M
6	Shyam	50	History	37/06/98	30000	M
7	Shivam	44	Computer	25/02/97	21000	M
8	Shalakha	33	Maths	31/07/97	20000	F

i) To show all information about the teacher of History department.
ii) To list the names of female teachers who are in Maths department.
iii) To list names of all teachers with their date of admission in ascending order.
iv) To insert a new row in the TEACHER table with the following data:
9,'Raja', 26,'Computer', {13/05/95}, 23000,'M'
v) Give the output of the following SQL statements.
a. Select COUNT(distinct department) from TEACHER;
b. Select MAX(Age) from TEACHER where SEX='F';
c. Select AVG(Salary) from TEACHER where SEX='M';

d. Select SUM(Salary) from TEACHER where DATOFJOIN<{12/07/96};

6. Consider the following tables EMPLOYEE and SALARIES. Write SQL commands for the statements (i) to (iv) and give outputs for SQL queries (v) to (viii)

TABLE: EMPLOYEE

EMPID	EMPNAME	SECTION	SEX	EXPERIENCE
101	Sabeena	PRODUCTION	F	10
102	Siva	ACCOUNTS	M	15
103	Ayaz	INSPECTION	M	10
104	Gayathri	PRODUCTION	F	8
105	Poorna	ACCOUNTS	F	9
106	Giridhar	INSPECTION	M	10

TABLE: SALARIES

EMPID	SALARY	INCENTIVE	ALLOWANCE
101	18000	1800	1325
103	23150	3100	1121
105	13120	2300	1240
106	12550	1250	1005

i. Display EMPNAME of all staff who are in “ACCOUNTS” having more than 10 years experience from the table EMPLOYEE**.**

ii. Display the average PAY of all staff working in “INSPECTION” department using the tables EMPLOYEE and SALARIES. The total pay is calculated as PAY= SALARY +INCENTIVE+ ALLOWANCE.

iii. Display the minimum INCENTIVE of female staff.

iv. Display the highest ALLOWANCE among all male staff.

v. SELECT count (*) from EMPLOYEE where SEX = “F”.

vi. SELECT EMPNAME, SECTION, ALLOWANCE from EMPLOYEE, SALARIES where SECTION = “ACCOUNTS” and EMPLOYEE.EMPID = SALARIES.EMPID.

vii. SELECT EMPNAME from EMPLOYEE where SECTION= INSPECTION and SEX=’M’

viii. Select count (distinct SECTION) from EMPLOYEE

7. Consider the following tables FACULTY and COURSES. Write SQL commands for the statements (i) to (iv) and give outputs for SQL queries (v) to (viii)

FACULTY

F_ID	Fname	Lname	Hire_date	Salary
102	Amit	Mishra	12-10-1998	12000
103	Nitin	Vyas	24-12-1994	8000
104	Rakshit	Soni	18-5-2001	14000

105	Rashmi	Malhotra	11-9-2004	11000
106	Sulekha	Srivastava	5-6-2006	10000
107	Niranjan	Kumar	26-8-1996	16000

COURSES

C_ID	F_ID	Cname	Fees
C21	102	Grid Computing	40000
C22	106	System Design	16000
C23	104	Computer Security	8000
C24	106	Human Biology	15000
C25	102	Computer Network	20000
C26	105	Visual Basic	6000
C27	107	Dreamweaver 4000	

(i) To display details of those Faculties whose date of joining is before 31-12-2001.
(ii) To display the details of courses whose fees is in the range of 15000 to 50000 (both values included).
(iii) To increase the fees of Dreamweaver course by 500.
(iv) insert new column in a COURSES named Level with String type..
(v) Select COUNT(DISTINCT F_ID) from COURSES;
(vi) Select MIN(Salary) from FACULTY,COURSES where COURSES.F_ID =FACULTY.F_ID;
(vii) Select SUM(Fees) from courses Group By F_ID having count(*) > 1;
(viii) Select Fname, Lname from FACULTY Where Lname like "M%";

8. Consider the following relation TEACHER and SALARY. Write SQL commands for the statement (i) to (iv) and output from (v) to (viii).

6

TABLE: TEACHER

TID	FIRST_NAME	LAST_NAME	ADDRESS	SUBJECT
010	Rohit	Sharma	83, Lok Vihar	English
105	Meeta	Rathi	842, Rajauri Garden	Physics
152	Seema	Verma	33, Safdarganj	Maths
215	Sarad	Singh	440, Ashok Vihar	Physics
244	Manish	Sengupta	24, New Street	Maths
300	Ram	Gupta	9, Fifth Road	Chemistry
355	Heena	Jain	12, Friends Street	Computer
400	Rachit	Sharma	10, Paschim Vihar	Computer
441	Punit	Jain	11, Rohini	Chemistry

TABLE: SALARY

TID	SARARY	BONUS	DESIGNATION
010	7500	1500	PGT
105	8500	1500	PGT
152	6000	1200	TGT
215	7500	1500	PGT

(i) To display FIRST_NAME, LAST_NAME and SUBJECT of all teachers of PHYSICS subject
(ii) To display all records in ascending order of LAST_NAME.
(iii) To display the TID, FIRST_NAME and total SALARY of all PGT from the table TEACHER and SALARY, where total salary is calculated as SALARY + BONUS.
(iv) To display the ADDRESS of teachers SUBJECT wise.
(v) SELECT FIRST_NAME, SALARY
FROM TEACHER, SALARY
WHERE DESIGNATION="TGT" AND TEACHER.TID = SALARY.TID;

(vi) SELECT DISTINCT(DESIGNATION), MAX (SALARY)
FROM SALARY
(vii)SELECT FIRST_NAME, ADDRESS
FROM TEACHER
WHERE FIRST_NAME LIKE ("S%");
(viii) SELECT SUM (BONUS)
FROM SALARY
WHERE DESIGNATION='PGT';

9. Write SQL commands for (a) to (j) and write output for (h) on the basis of ***Teacher*** relation given below.

No	Name	Age	Department	Date of Join	Salary	Sex
1.	Jigal	34	Computer	10/01/97	12000	M
2.	Sharmila	31	History	24/03/98	20000	F
3.	Sandeep	32	Maths	12/12/96	30000	M
4.	Sangeeta	35	History	01/07/99	40000	F
5.	Rakesh	42	Maths	05/09/97	25000	M
6.	Shyam	50	History	27/02/97	30000	M
7.	Shiv Om	44	Computer	25/02/97	21000	M
8.	Shalakha	33	Maths	31/07/97	20000	F

I. To show all information about the teacher of history department.
II. To list the names of female teachers who are in Maths department
III. To list names of all teachers with their date of joining in ascending order.
IV. To count the number of teachers with age<35.
V. To insert a new row in the TEACHER table with the following data:
9,"Raja",26,"Computer",'13/05/95',2300,"M".
VI. To count the number of teachers having salary >=12000, with each department.

10. Given the following **LAB** table, write SQL command for the questions (i) to (iii) and give the output of (iv). **6**

LAB

No	ItemName	CostPerItem	Quantity	Dateofpurchase	Warranty	Operational
1	Computer	60000	9	21/5/96	2	7
2	Printer	15000	3	21/5/97	4	2
3	Scanner	18000	1	29/8/98	3	1
4	Camera	21000	2	13/10/96	1	1
5	Switch	8000	1	31/10/99	2	1
6	UPS	5000	5	21/5/96	1	4
7	Router	25000	2	11/1/2000	2	5

(i) To select the ItemName,which are within the Warranty period till present date.
(ii) To display all the itemName whose name starts with "C".
(iii) To list the ItemName in ascending order of the date of purchase where quantity is more than 3.
(iv) Give the output of the following SQL commands:
(a) select min(DISTINCT Quantity) from LAB; (b) select max(Warranty) from LAB;
(c) select sum(CostPerItem) from Lab;

11. Write the SQL commands for (i) to (iv) and outputs for (v) to (viii) on the basis of tables BOOKS and ISSUES.(4 X 1 + 4 X 0.5)

Table: ISSUES

Book_ID	Qty_Issued
L02	13
L04	5
L05	21

Table: BOOKS

Book_ID	BookName	AuthorName	Publisher	Price	Qty
L01	Maths	Raman	ABC	70	20
L02	Science	Agarkar	DEF	90	15
L03	Social	Suresh	XYZ	85	30
L04	Computer	Sumita	ABC	75	7
L05	Telugu	Nannayya	DEF	60	25
L06	English	Wordsworth	DEF	55	12

(i) To show Book name, Author name and Price of books of ABC publisher.
(ii) To display the details of the books in descending order of their price.
(iii) To decrease the Qty_Issued from ISSUES table by 3 (all rows must decrease).
(iv) To display the Book Id, Book name, Publisher, Price, Qty, Qty_Issued from both the tables with their matching Book ID.
(v) SELECT sum(price) FROM Books WHERE Publisher = “DEF”;
(vi)SELECT Publisher, min(price) FROM Books GROUP BY Publisher;
(vii)SELECT Price from Books, Issues where Books.Book_ID=Issues.Book_ID AND Qty_Issued=5;
(viii)SELECT Count(Distinct Publisher) FROM Books;

UNIT 4 : Boolean Algebra : 1/2 Marks Questions

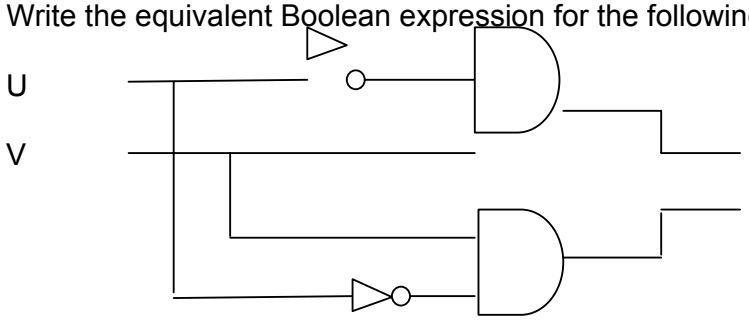
1. Define Binary logic ?
2. What is a Boolean Operation ?
3. Define a boolean function ?
4. Define a Boolean Expression ?
5. Name the three primary and secondary operators of Boolean Algebra ?State any four postulates of boolean algebra ?
6. Define Idempotent Law ?
7. Define Absorptive Law ?
8. Define Involution Law ?
9. What is De Morgan's Theorem ?
10. State the principle of duality ?
11. State the steps required to calculate the dual of any expression ?
12. State the dual of : A+A' = 1
13. What is a Boolean Function ?
14. Define the Sum Of Products format of a boolean expression ?
15. Define the Product of Sums format of a boolean expression ?
16. What is a Karnaugh map ?
17. Draw the truth table of NAND gate ?
18. Define the XNOR gate ?
19. What is a Half Adder ?
20. What is a Full Adder ?
21. Differentiate between an Encoder and a Decoder ?
22. What are Universal Gates ? Name any two Universal Gates ?
23. Define the working of a XOR gate ?
24. What is a Multiplexer ?
25. What is a Multivibrator ?
26. What is a Minterm ?
27. What is a Maxterm ?
28. What is a Canonical Sum of Products ?
29. What is a Canonical Product of Sums ?
30. State the total number of combinations possible for a three input gate ?
31. Draw a logical circuit diagram for the following Boolean expression: A.(B+C)'
32. Convert the following Boolean expression into its equivalent Canonical Sum of Products Form (U'+V'+W'). (U+V'+W'). (U+V+W)
33. Draw the Logical Circuit Diagram for the following Boolean Expression: (A'.B').+(C.D')
34. Write the equivalent Canonical Product of Sum for the following expression.
    F(A,B,C) = ∑(1,2,3,4)

35. Write the SOP form of a Boolean function G, which is represented in a truth table as follows:

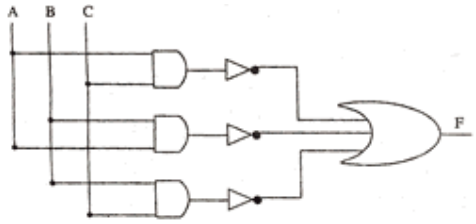
P	Q	R	f(G)
0	0	0	1
0	0	1	0
0	1	0	0

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

36. Write the equivalent Boolean expression for the following Logic Circuit:



37. Write the equivalent Boolean expression for the following Circuit

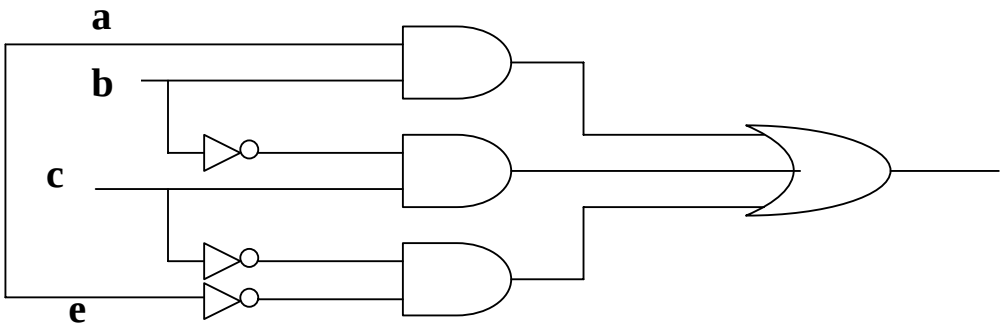


38. For the given truth table, give canonical sum-of-products(SOP) and canonical product-of-sum (POS) expression

X	Y	Z	F o/p
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

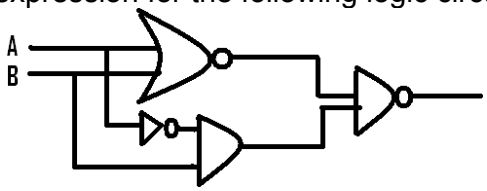
39. Write the principal of Duality and write the dual of the Boolean Expression:
(B` + C) + A`

40. Interpret the following logical circuit as Boolean expression.



42. Prove that **XY+YZ+YZ'=Y**

43. Write the equivalent Boolean expression for the following logic circuit.



44. Express the F(X,Z)=X+X'Z into canonical SOP form.

45. Write the equivalent canonical POS expression for the following SOP expression:
F(x,y,z)=Σ(0,2,5,6)

Boolean Algebra : 3/4 Marks Questions

1. Reduce the following Boolean expression using K-map.

   F(A, B, C, D)= Σ(0, 1, 2, 4, 5, 7, 8, 9, 10, 11, 14)

2. Reduce the following Boolean expression using the K-map. (3)
   F(A,B,C,D)= Σ(0,1,3,4,7,8,11,12,15);
3. If **F(a,b,c,d)= Σ (1,3,4,5,7,9,11,12,13,15)** obtain the simplified form using K-Map
4. Reduce the following Boolean expression using K-map:
   H(U, V, W, Z) = Σ (0, 1, 4, 5, 6, 7, 10, 11, 12, 13, 14, 15)

5. Reduce the following Boolean expression using K-map:
   H(U, V, W, Z) = Σ (0, 1, 4, 5, 6, 7, 10, 11, 12, 13, 14, 15)
6. Reduce the following Boolean expression using K-Map
   F(A,B,C,D) = ∑ (0,1,3,4,5,7,8,9,11,12,13,15) obtain the simplified form using K-Map.
7. Reduce the following Boolean expression using K-map

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

UNIT 5 : Communication and Network Concepts : 1 OR 2 Mark Questions

1. Differentiate between Internet and Intranet.
2. Expand the following
   (i) CDMA (ii) URL
3. What is a Bandwidth?
4. Expand the following terms with respect to networking:
   (i) CDMA (ii) SMS (iii) WLL (iv) XML
5. How is a BUS TOPOLOGIES different from A STAR TOPOLOGIES ?
6. Explain FLOSS
7. What is FREEWARE? How is it different from other OSS

8. what are Cookies?
9. Expand the following terminology – i) PHP ii) IMAP
10. Differentiate between routers and bridge.
11. Expands the following terms:
    (i) EDGE (ii) XML
12. For what purpose Mozilla is used?
13. What is the function of TCP protocol?
14. Write the different type of Topologies with one advantage and one disadvantage.
15. Define Circuit Switching and Packet Switching
16. Give one advantage and one disadvantage of optical fiber and coaxial cable used in communication.
17. Explain the following terms
    (1) Video Conferencing
    (2)TCP\IP
18. Explain function of hub and router.
19. Expand the following terms:
    (i) URL (ii) ISP (iii) DHTML (iv) CDMA
20. Differentiate between message switching and packet switching
21. Expand the following terms: (i) GSM (ii)TCP/IP
22. What is the difference between XML and HTML? Write two differences.
23. What is a topology? Write short note on Bus topology.
24. What are the advantages of Ring Topology?
25. Explain Freeware and Shareware?
26. What is Cyber Crime?

4 Marks Questions : Communication and Network Concepts

1. Knowledge Supplement Organization has set up its new centre at Manglore for its office and web based activities. It has four buildings as shown in the diagram below:

Center to center distance between various buildings

Alpha to Beta	50m
Beta to Gamma	150m
Gamma to Lambda	25m
Alpha to Lambda	170m
Beta to Lambda	125m
Alpha to Gamma	90m

Number of Computers

Alpha	25
Beta	50
Gamma	125
Lambda	10

i) Suggest a cable layout of connections between the buildings
ii) Suggest the most suitable place(i.e building) to house the server of this organization with a suitable reason.
iii) Suggest the placement of the following devices with justification:
   i. Repeater
   ii. Hub/Switch
iv) The organization is planning to link its front office situated in the city in a hilly region where cable connection is not feasible, suggest an economic way to connect it with reasonably high speed?

2. **BHARATH ELECTRONICS COMPANY** in Coimbatore is setting up the network between its different departments located in different blocks. There are 4 blocks named as Meera (M), Tagore (T), Kalidas (K) and Bharathi (B).
Distances between various blocks are given below:

Block B to Block K	100 m
Block B to Block M	200 m
Block B to Block T	400 m
Block K to Block M	300 m
Block M to Block P	100m
Block R to Block P	450 m

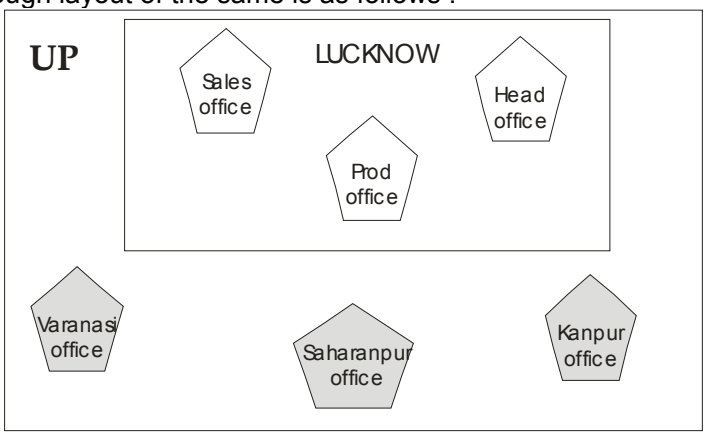
a. **Number of Computers:**

Block M	15
Block R	100
Block A	50
Block P	150

i. Suggest a suitable Topology for networking the computers of all Blocks.
ii. Name the Block where the Server is to be installed. Justify your answer.
iii. Suggest the placement of Hub/Switch in the network.

iv. Mention an economic technology to provide Internet accessibility to allBlocks.

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



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

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

Number of computers :

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

i) Suggest the placement of the ***repeater*** with justification. Name the branch where the ***server*** should be installed. Justify your answer.

ii) Suggest the ***device*** to be procured by the company for connecting all the computers within each of its offices out of the following devices:

- Modem
- Telephone
- Switch/Hub

iv) The company is planning to link its head office situated in Lucknow with the office at Saharanpur. Suggest an economic way to connect it; the company is ready to compromise on the speed of connectivity. Justify your answer.

4. Dr. Rizvi Education Society of India is starting its new CBSE School in Mumbai (Maharashtra). The society is already running a School in Jaunpur (UP) named Dr. Rizvi Learners' Academy, having 3 major buildings in 2 km area campus. As a network expert you need to suggest the network plan as per E1 to E4:
Wire Distance Between Various Buildings:

Library building to Admin building	90m
Library building to Academic building	80m
Academic building to Admin building	15m
Jaunpur School to Mumbai School	1350km

Expected number of Computers to be installed in various buildings:

Library Building	20
Academic building	150
Admin building	35
Mumbai School	5

E1. Suggest the cable layout among various buildings inside school campus for connecting the buildings.

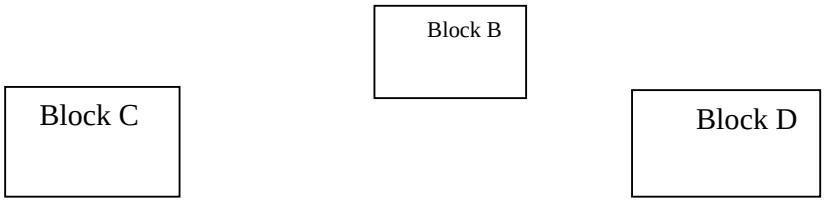
E2. Suggest the most suitable place to house the server of the school with a suitable reason.

E3. Suggest an efficient device from the following to be installed in each of the building to connect all the computers:

(i) Bridge (ii) Repeater (iii) Switch

E4. Suggest the most suitable service (very high speed) to provide data connectivity between Rizvi Learners' in Jaunpur and Mumbai CBSE School from the options

5. Knowledge Supplement Organization has set up its new center at Mangalore for its office and web based activities. It has 4 blocks of buildings as shown in the diagram below.



The distances between the building are as :
Block A to Block C- 120 meters
Block A to Block B- 20 meters
Block A to Block D- 550 meters
Block B to Block D- 80 meters
Block D to Block C- 110 meters
Block B to Block C- 280 meters

The number of computers in each Block are as follows:
Block A - 120
Block B - 180
Block C - 20
Block D - 110

(i) Suggest a cable layout of connections between the blocks and type of cable.

(ii) Suggest the most suitable place (i.e. block) to house the server of this organization with a suitable reason.

(iii) Suggest the placement of the following devices with justification.
(a) Repeater
(b) Hub/Switch

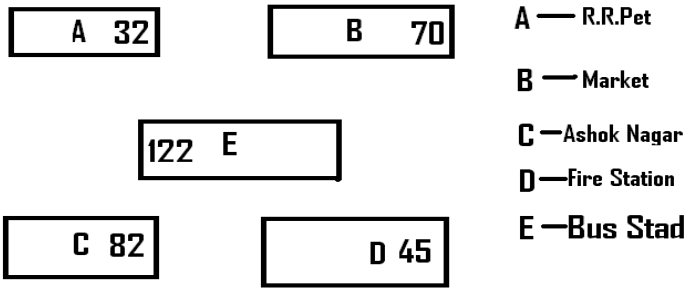
(iv) The organization is planning to link its front office situated in the city in a Hilly region where cable connection is not feasible, suggest an economic way to connect it with reasonably high speed.

6. In Hyderabad, 5 ABC Bank branches are available. One is at RR Pet, other at Market, other at Ashok Nagar, other at Fire Station and the last one at Bus Stand. Higher official want to keep a network between these 5 branches. The branch names(A to E) and the number of computers

7.

in each branch(given inside the rectangle) is given below.

Distance between various buildings	
A to B	50 Mts
B to C	30 Mts
C to D	30 Mts
D to E	35 Mts
E to C	40 Mts
D to A	120 Mts
D to B	45 Mts
E to B	65 Mts



(i) Suggest a possible cable layout for connecting the buildings.
(ii)Suggest the most suitable place to install the server of this organization with a suitable reason
(iii) Suggest the placement of the following devices with justification.
(a) Hub/Switch (b) Modem
(iv) The Bank wans to link its head Office in 'A' building to its main office at Mumbai
(a) Which type of transmission medium is appropriate for such a link?
(b) What type of network this connection result into?

Computer Science (Code 083)
Sample Paper Set - 1

Max. Marks: 70

Duration: 3 Hours

1.

(a) What is the difference between Global Variable and Local Variable? 2

(b) Write the names of the header files to which the following belong: 1
(i) strcmp() (ii) fabs()

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

```
#include [iostream.h]
class PAYITNOW
{
   int Charge;
PUBLIC:
    void Raise(){cin>>Charge;}
    void Show{cout<<Charge;}
};
void main()
{
   PAYITNOW P;
   P.Raise();
   Show();
}
```

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

```
#include <iostream.h>
struct PLAY
{ int Score, Bonus;};
void Calculate(PLAY &P, int N=10)
{
     P.Score++;P.Bonus+=N;
}
void main()
{
     PLAY PL={10,15};
     Calculate(PL,5);
     cout<<PL.Score<<":"<<PL.Bonus<<endl;
     Calculate(PL);
     cout<<PL.Score<<":"<<PL.Bonus<<endl;
     Calculate(PL,15);
     cout<<PL.Score<<":"<<PL.Bonus<<endl;
}
```

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

```
#include <iostream.h>
#include <ctype.h>
void Encrypt(char T[])
{
     for (int i=0;T[i]!='\0';i+=2)
```

```
            if (T[i]=='A' || T[i]=='E') T[i]='#';
            else if (islower(T[i])) T[i]=toupper(T[i]);
                        else T[i]='@';
}
void main()
{
      char Text[]="SaVE EArtH";//The two words in the string Text
                  //are separated by single space
      Encrypt(Text);
      cout<<Text<<endl;
}
```

(f) In the following program, if the value of N given by the user is 15, what maximum and minimum values the program could possibly display? **2**

```
#include <iostream.h>
#include <stdlib.h>
void main()
{
     int N,Guessme;
     randomize();
     cin>>N;
     Guessme=random(N)+10;
     cout<<Guessme<<endl;
}
```

2.

(a) What do you understand by Data Encapsulation and Data Hiding? **2**

a) Answer the questions (i) and (ii) after going through the following class: **2**

```
class Seminar
{
     int Time;
public:
     Seminar()                          //Function 1
     {
     Time=30;cout<<"Seminar starts now"<<end1;
   }
   void Lecture()                       //Function 2
     {
     cout<<"Lectures in the seminar on"<<end1;
   }
     Seminar(int Duration)         //Function 3
     {
     Time=Duration;cout<<"Seminar starts now"<<end1;
   }
   ~Seminar()                           //Function 4
   {
     cout<<"Vote of thanks"<<end1;
   }
};
```

i) In Object Oriented Programming, what is **Function 4** referred as and when does it get invoked/called?

ii) In Object Oriented Programming, which concept is illustrated by **Function 1** and **Function 3** together? Write an example illustrating the calls for these functions.

(c) Define a class TEST in C++ with following description: **4**

Private Members

a. **TestCode of type integer**
b. **Description of type string**
c. **NoCandidate of type integer**
d. **CenterReqd (number of centers required) of type integer**
e. **A member function CALCNTR() to calculate and return the number of centers as (NoCandidates/100+1)**

Public Members

- **A function SCHEDULE() to allow user to enter values for TestCode, Description, NoCandidate & call function CALCNTR() to calculate the number of Centres**
- **A function DISPTEST() to allow user to view the content of all the data members**

(d) Answer the questions (i) to (iv) based on the following: **4**

```
class PUBLISHER
{
     char Pub[12];
     double Turnover;
protected:
     void Register();
public:
     PUBLISHER();
     void Enter();
     void Display();
};

class BRANCH
{
     char CITY[20];
protected:
     float Employees;
public:
     BRANCH();
     void Haveit();
     void Giveit();
};

class AUTHOR:private BRANCH,public PUBLISHER
{
     int Acode;
   char Aname[20];
     float Amount;
public:
     AUTHOR();
     void Start();
     void Show();
};
```

(i) Write the names of data members, which are accessible from objects belonging to class AUTHOR.

(ii) Write the names of all the member functions which are accessible from objects belonging to class BRANCH.

(iii) Write the names of all the members which are accessible from member functions of class AUTHOR.

(iv) How many bytes will be required by an object belonging to class AUTHOR?

3.

(a) Write a function in C++ to merge the contents of two sorted arrays A & B into third array C. Assuming array A is sorted in ascending order, B is sorted in descending order, the resultant array is required to be in ascending order. **4**

(b) An array S[40][30] is stored in the memory along the row with each of the element occupying 2 bytes, find out the memory location for the element S[20][10], if an element S[15][5] is stored at the memory location 5500. **4**

(c) Write a function in C++ to perform Insert operation in a dynamically allocated Queue containing names of students. **4**

(d) Write a function in C++ to find the sum of both left and right diagonal elements from a two dimensional array (matrix). **2**

(e) Evaluate the following postfix notation of expression: **2**
20,30,+,50,40,-,*

4.

(a) Observe the program segment given below carefully and fill the blanks marked as Statement 1 and Statement 2 using seekp() and seekg() functions for performing the required task. **1**

```
#include <fstream.h>
class Item
{
   int Ino;char Item[20];
public:
   //Function to search and display the content from a particular
   //record number
   void Search(int );
   //Function to modify the content of a particular record number
   void Modify(int);
};
void Item::Search(int RecNo)
{
     fstream File;
     File.open("STOCK.DAT",ios::binary|ios::in);
     _____________________
     //Statement 1
     File.read((char*)this,sizeof(Item));
     cout<<Ino<<"==>"<<Item<<endl;
     File.close();
}
void Item::Modify(int RecNo)
```

```
{
      fstream File;
      File.open("STOCK.DAT",ios::binary|ios::in|ios::out);
      cout>>Ino;cin.getline(Item,20);
                  _____________________
//Statement 2
      File.write((char*)this,sizeof(Item));
      File.close();
}
```

(b) Write a function in C++ to count the number of lines present in a text file "STORY.TXT". 2

(c) Write a function in C++ to search for a BookNo from a binary file "BOOK.DAT", assuming the binary file is containing the objects of the following class. 3

```
class BOOK
{
      int Bno;
      char Title[20];
public:
      int RBno(){return Bno;}
      void Enter(){cin>>Bno;gets(Title);}
      void Display(){cout<<Bno<<Title<<endl;}
};
```

5.

(a) What do you understand by Degree and Cardinality of a table? 2

(b) Consider the following tables ACTIVITY and COACH. Write SQL commands for the statements (i) to (iv) and give outputs for SQL queries (v) to (viii) 6

Table: ACTIVITY

ACode	ActivityName	ParticipantsNum	PrizeMoney	ScheduleDate
1001	Relay 100x4	16	10000	23-Jan-2004
1002	High jump	10	12000	12-Dec-2003
1003	Shot Put	12	8000	14-Feb-2004
1005	Long Jump	12	9000	01-Jan-2004
1008	Discuss Throw	10	15000	19-Mar-2004

Table: COACH

PCode	Name	ACode
1	Ahmad Hussain	1001
2	Ravinder	1008
3	Janila	1001
4	Naaz	1003

(i) To display the name of all activities with their Acodes in descending order.

(ii) To display sum of PrizeMoney for each of the Number of participants groupings (as shown in column ParticipantsNum 10,12,16)

(iii) To display the coach's name and ACodes in ascending order of ACode from the table COACH

(iv) To display the content of the GAMES table whose ScheduleDate earliar than 01/01/2004 in ascending order of ParticipantNum.

(v) SELECT COUNT(DISTINCT ParticipantsNum) FROM ACTIVITY;

(vi)SELECT MAX(ScheduleDate),MIN(ScheduleDate) FROM ACTIVITY;

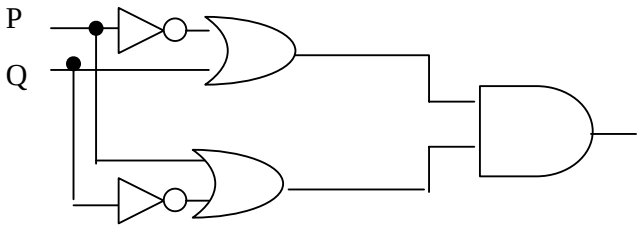
(vii) SELECT SUM(PrizeMoney) FROM ACTIVITY;

(viii) SELECT DISTINCT ParticipantNum FROM COACH;

6.

(a) State and verify Demorgan's Laws. 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 in a truth table as follows: 1

U	V	W	F
0	0	0	1
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

(d) Reduce the following Boolean Expression using K-Map: 3
F(A,B,C,D)=Σ(0,1,2,4,5,6,8,10)

7.

a) What is the significance of ARPANET in the network? 1

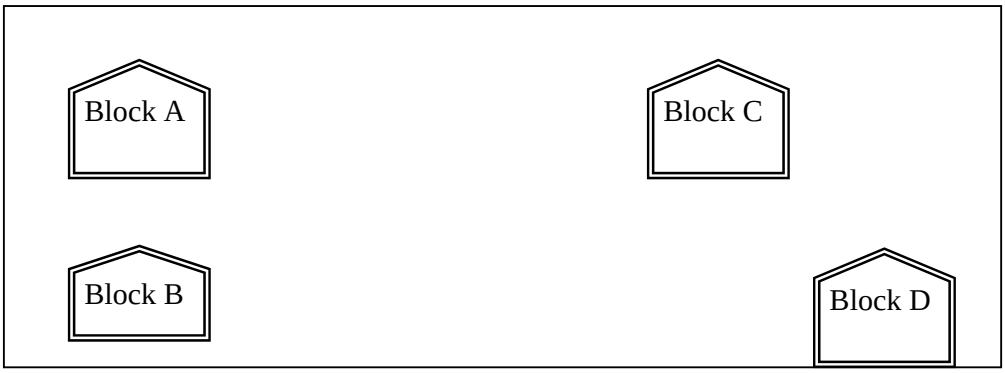
b) Expand the following terminologies: 1

(i) CDMA (ii) GSM

c) Give two major reasons to have network security. 1

d) What is the purpose of using a Web Browser? Name any one commonly used Web Browser. 1

e) Knowledge Supplement Organisation has set up its new center at Mangalore for its office and web based activities. It has 4 blocks of buildings as shown in the diagram below:



Center to center distances between various blocks

Black A to Block B	50 m
Block B to Block C	150 m
Block C to Block D	25 m
Block A to Block D	170 m
Block B to Block D	125 m
Block A to Block C	90 m

Number of Computers

Black A	25
Block B	50
Block C	125
Block D	10

e1) Suggest a cable layout of connections between the blocks. 1

e2) Suggest the most suitable place (i.e. block) to house the server of this organisation with a suitable reason. 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 front office situated in the city in a hilly region where cable connection is not feasible, suggest an economic way to connect it with reasonably high speed? 1

Computer Science (Code 083)
Sample Paper Set –II

Max. Marks: 70 **Duration: 3 Hours**

1.

(a) What is the difference between Object Oriented Programming and Procedural Programming? **2**

(d) Write the names of the header files to which the following belong: **1**
(i) frexp() **(ii) isalnum()**

(e) Rewrite the following program after removing the syntactical errors (if any). Underline each correction. **2**

```
#include <iostream.h>
struct Pixels
{    int Color,Style;}
void ShowPoint(Pixels P)
{    cout<<P.Color,P.Style<<endl;}
void main()
{
     Pixels Point1=(5,3);
     ShowPoint(Point1);
   Pixels Point2=Point1;
     Color.Point1+=2;
   ShowPoint(Point2);
}
```

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

```
#include <iostream.h>
void Changethecontent(int Arr[], int Count)
{
        for (int C=1;C<Count;C++)
                Arr[C-1]+=Arr[C];
}
void main()
{
        int A[]={3,4,5},B[]={10,20,30,40},C[]={900,1200};
        Changethecontent(A,3);
        Changethecontent(B,4);
        Changethecontent(C,2);
        for (int L=0;L<3;L++) cout<<A[L]<<'#';
        cout<<endl;
        for (L=0;L<4;L++) cout<<B[L] <<'#';
        cout<<endl;
        for (L=0;L<2;L++) cout<<C[L] <<'#';
}
```

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

```
#include <iostream.h>
struct Game
{
        char Magic[20];int Score;
```

```
};
void main()
{
        Game M={"Tiger",500};
        char *Choice;
        Choice=M.Magic;
        Choice[4]='P';
        Choice[2]='L';
        M.Score+=50;
        cout<<M.Magic<<M.Score<<endl;
  Game N=M;
        N.Magic[0]='A';N.Magic[3]='J';
        N.Score-=120;
        cout<<N.Magic<<N.Score<<endl;
}
```

(g) In the following program, if the value of N given by the user is 20, what maximum and minimum values the program could possibly display? 2

```
#include <iostream.h>
#include <stdlib.h>
void main()
{
     int N,Guessnum;
     randomize();
     cin>>N;
     Guessnum=random(N-10)+10;
     cout<<Guessnum<<endl;
}
```

2.

a) What do you understand by Polymorphism? Give a suitable example of the same. 2

b) Answer the questions (i) and (ii) after going through the following program: 2

```
class Match
{
     int Time;
public:
     Match()                              //Function 1
     {
     Time=0;
     cout<<"Match commences"<<end1;
   }
   void Details()                         //Function 2
     {
     cout<<"Inter Section Basketball Match"<<end1;
   }

     Match(int Duration)      //Function 3
     {
     Time=Duration;
     cout<<"Another Match begins now"<<end1;
   }
   Match(Match &M)        //Function 4
```

```
        {
        Time=M.Duration;
        cout<<"Like Previous Match "<<end1;
      }

};
```

iii) **Which category of constructor - Function 4 belongs to and what is the purpose of using it?**

iv) **Write statements that would call the member Functions 1 and 3**

c) **Define a class in C++ with following description:** **4**

Private Members

a. **A data member Flight number of type integer**
b. **A data member Destination of type string**
c. **A data member Distance of type float**
d. **A data member Fuel of type float**
e. **A member function CALFUEL() to calculate the value of Fuel as per the following criteria**

Distance	Fuel
<=1000	500
more than 1000 and <=2000	1100
more than 2000	2200

Public Members

- **A function FEEDINFO() to allow user to enter values for Flight Number, Destination, Distance & call function CALFUEL() to calculate the quantity of Fuel**
- **A function SHOWINFO() to allow user to view the content of all the data members**

d) **Answer the questions (i) to (iv) based on the following:** **4**

```
class CUSTOMER
{
     int Cust_no;
     char Cust_Name[20];
protected:
   void Register();
public:
     CUSTOMER();
     void Status();
};
class SALESMAN
{
     int Salesman_no;
     char Salesman_Name[20];
protected:
     float Salary;
public:
     SALESMAN();
     void Enter();
     void Show();
};
class SHOP : private CUSTOMER , public SALESMAN
{
```

```
        char Voucher_No[10];
        char Sales_Date[8];
public:
        SHOP();
        void Sales_Entry();
        void Sales_Detail();
};
```

(vi) Write the names of data members which are accessible from objects belonging to class CUSTOMER.

(vii) Write the names of all the member functions which are accessible from objects belonging to class SALESMAN.

(viii) Write the names of all the members which are accessible from member functions of class SHOP.

(ix) How many bytes will be required by an object belonging to class SHOP?

3.

(a) Write a function in C++ to combine the contents of two equi-sized arrays A and B by computing their corresponding elements with the formula 2*A[i]+3*B[i]; where value i varies from 0 to N-1 and transfer the resultant content in the third same sized array. 4

(f) An array P[20][30] is stored in the memory along the column with each of the element occupying 4 bytes, find out the memory location for the element P[5][15], if an element P[2][20] is stored at the memory location 5000. 4

(g) Write a function in C++ to perform Push operation on a dynamically allocated Stack containing real numbers. 4

(h) Write a function in C++ to find sum of rows from a two dimensional array. 2

(i) Evaluate the following postfix notation of expression: 2
True, False, AND, True, True, NOT, OR, AND

4.

(a) Observe the program segment given below carefully and fill the blanks marked as Statement 1 and Statement 2 using seekg() and tellg() functions for performing the required task. 1

```
#include <fstream.h>
class Employee
{
   int Eno;char Ename[20];
public:
   //Function to count the total number of records
   int Countrec();
};
int Item::Countrec()
{
        fstream File;
        File.open("EMP.DAT",ios::binary|ios::in);
        _____________________
        //Statement 1
```

```
        int Bytes =          _____________________
        //Statement 2

        int Count = Bytes / sizeof(Item);
        File.close();
        return Count;
}
```

(b) Write a function in C++ to count the number of alphabets present in a text file "NOTES.TXT". 2

(c) Write a function in C++ to add new objects at the bottom of a binary file "STUDENT.DAT", assuming the binary file is containing the objects of the following class. 3

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

void Addnew()
{
        fstream FIL;
        FIL.open("STUDENT.DAT",ios::binary|ios::app);
        STUD S;
        char CH;
        do
        {
                S.Enter();
                FIL.write((char*)&S,sizeof(S));
                cout<<"More(Y/N)?";cin>>CH;
        }
        while(CH!='Y');
        FIL.close();
}
```

5.

(a) What do you understand by Primary Key & Candidate Keys? 2

(b) Consider the following tables GAMES and PLAYER. Write SQL commands for the statements (i) to (iv) and give outputs for SQL queries (v) to (viii) 6

Table: GAMES

GCode	GameName	Number	PrizeMoney	ScheduleDate
101	Carom Board	2	5000	23-Jan-2004
102	Badminton	2	12000	12-Dec-2003
103	Table Tennis	4	8000	14-Feb-2004
105	Chess	2	9000	01-Jan-2004
108	Lawn Tennis	4	25000	19-Mar-2004

Table: PLAYER

PCode	Name	Gcode
1	Nabi Ahmad	101
2	Ravi Sahai	108
3	Jatin	101
4	Nazneen	103

(i) To display the name of all Games with their Gcodes

(ii) To display details of those games which are having PrizeMoney more than 7000.

(iii)To display the content of the GAMES table in ascending order of ScheduleDate.

(iv) To display sum of PrizeMoney for each of the Number of participation groupings (as shown in column Number 2 or 4)

(x) SELECT COUNT(DISTINCT Number) FROM GAMES;

(vi)SELECT MAX(ScheduleDate),MIN(ScheduleDate) FROM GAMES;

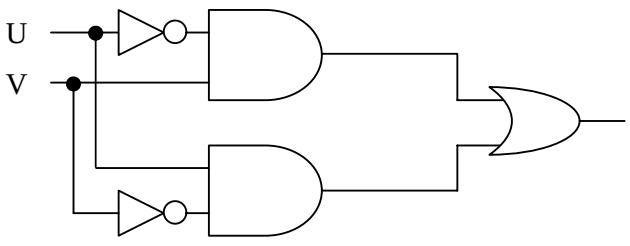
(vii) SELECT SUM(PrizeMoney) FROM GAMES;

(viii) SELECT DISTINCT Gcode FROM PLAYER;

6.

(a) State and algebraically verify Absorbtion Laws. 2

(b) Write the equivalent Boolean Expression for the following Logic Circuit 2



(e) Write the SOP form of a Boolean function G, which is represented in a truth table as follows: **1**

P	Q	R	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

(f) Reduce the following Boolean Expression using K-Map: **3**
F(U,V,W,Z)=Π(0,1,2,4,5,6,8,10)

7.

a) Define the term Bandwidth. Give unit of Bandwidth. **1**

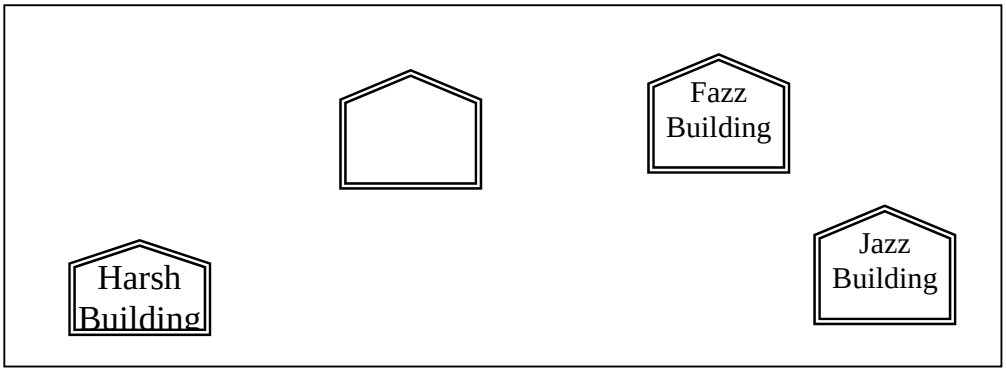
b) Expand the following terminologies: **1**
(i) HTML (ii) XML

c) Define the term firewall. **1**

d) What is the importance of URL in networking? **1**

e)

Ravya Industries has set up its new center at Kaka Nagar for its office and web based activities. The company compound has 4 buildings as shown in the diagram below:



Center to center distances between various buildings is as follows:

Harsh Building to Raj Building	50 m
Raz Building to Fazz Building	60 m
Fazz Building to Jazz Building	25 m
Jazz Building to Harsh Building	170 m
Harsh Building to Fazz Building	125 m
Raj Building to Jazz Building	90 m

Number of Computers in each of the buildings is follows:

Harsh Building	15
Raj Building	150
Fazz Building	15
Jazz Bulding	25

e1) Suggest a cable layout of connections between the buildings. 1

e2) Suggest the most suitable place (i.e. building) to house the server of this organisation with a suitable reason. 1

e3) Suggest the placement of the following devices with justification: 1

(i) Internet Connecting Device/Modem

(ii) Switch

e4) The organisation is planning to link its sale counter situated in various parts of the same city, which type of network out of LAN, MAN or WAN will be formed? Justify your answer. 1

COMPUTER SCIENCE (Theory) - Class XII Sample Question

Paper–I

Subject Code - 083

TIME : 3 Hrs **MM : 70**

No.	Questions	Marks
1.		
(a)	**What is the difference between Global Variable and Local Variable? Also, give a suitable C++ code to illustrate both.**	2

(b)	**Which C++ header file(s) will be essentially required to be included to run / execute the following C++ code:**
	void main()
	{
	char Msg[ ]="Sunset Gardens";
	for (int I=5;I<strlen(Msg);I++)
	puts(Msg);
	}
		1

(c)	**Rewrite the following program after removing the syntactical errors (if any). Underline each correction.**
	#include [iostream.h]
	class MEMBER
	{
	int Mno;float Fees;
	PUBLIC:
	void Register(){cin>>Mno>>Fees;}
	void Display{cout<<Mno<<" : "<<Fees<<endl;}
	};
	void main()
	
	{
	MEMBER M;
	Register();
	M.Display();
	}
		2

No.	Questions	Marks

(d)	**Find the output of the following program:**
	#include <iostream.h>
	struct GAME
	{ int Score, Bonus;};
	void Play(GAME &g, int N=10)
	{
	g.Score++;g.Bonus+=N;
	}
	void main()
	{
	GAME G={110,50}; Play(G,10);
	cout<<G.Score<<":"<<G.Bonus<<endl;
	Play(G);
	cout<<G.Score<<":"<<G.Bonus<<endl;
	Play(G,15);
	cout<<G.Score<<":"<<G.Bonus<<endl;
	}
		3

(e)	**Find the output of the following program:**
	#include <iostream.h>
	void Secret(char Str[ ])
	{
	for (int L=0;Str[L]!='\0';L++);
	for (int C=0;C<L/2;C++)
	if (Str[C]=='A' || Str[C]=='E')
	Str[C]='#';
	else
	{
	char Temp=Str[C];
		2

No.	Questions	Marks

	Str[C]=Str[L-C-1];
	Str[L-C-1]=Temp;
	}
	}
	void main()
	{
	char Message[ ]="ArabSagar";
	Secret(Message);
	cout<<Message<<endl;
	}	

(f)	**In the following program, if the value of Guess entered by the user is 65, what will be the expected output(s) from the following options (i), (ii), (iii) and (iv)?**
	#include <iostream.h>
	#include <stdlib.h>
	void main()
	{
	int Guess;
	randomize();
	cin>>Guess;
	for (int I=1;I<=4;I++)
	{ New=Guess+random(I);
	cout<<(char)New;
	}
	}
	(i) ABBC
	(ii) ACBA
	(iii) BCDA
	(iv) CABD
		2

No.	Questions	Marks
2.		
(a)	**What do you understand by Data Encapsulation and Data Hiding? Also, give a suitable C++ code to illustrate both.**	2
(b)	**Answer the questions (i) and (ii) after going through the following class:**	2

```
class Seminar
{
int Time;
public:
Seminar()                                   //Function 1
{
Time=30;cout<<"Seminar starts now"<<end1;
}
void Lecture()                              //Function 2
{
cout<<"Lectures in the seminar on"<<end1;
}
Seminar(int Duration)                       //Function 3
{
Time=Duration;cout<<"Seminar starts now"<<end1;
}
~Seminar()
                                            //Function 4
{
cout<<"Vote of thanks"<<end1;
}
};
```

i) In Object Oriented Programming, what is Function 4 referred as and when does it get invoked/called?

ii) In Object Oriented Programming, which concept is illustrated by Function 1 and Function 3 together? Write an example illustrating the calls for these functions.

No.	Questions	Marks
(c)	**Define a class TEST in C++ with following description: Private Members** • TestCode of type integer • Description of type string • NoCandidate of type integer • CenterReqd (number of centers required) of type integer • A member function CALCNTR() to calculate and return the number of centers as (NoCandidates/100+1) **Public Members** • A function SCHEDULE() to allow user to enter values for TestCode, Description, NoCandidate & call function CALCNTR() to calculate the number of Centres • A function DISPTEST() to allow user to view the content of all the data members	4

(d)	**Answer the questions (i) to (iv) based on the following:**
	class PUBLISHER
	{
	char Pub[12];
	double Turnover;
	protected:
	void Register();
	public:
	PUBLISHER();
	void Enter(); void
	Display();
	};
	class BRANCH
	{
	char CITY[20];
	protected:
	float Employees;
		4

No.		Questions	Marks

		public:
		BRANCH();
		void Haveit();
		void Giveit();
		};
		class AUTHOR : private BRANCH , public PUBLISHER
		{
		int Acode;
		char Aname[20];
		float Amount; public:
		AUTHOR();
		void Start();
		void Show();
		};	

	(i)	Write the names of data members, which are accessible from objects belong- ing to class AUTHOR.	
	(ii)	Write the names of all the member functions which are accessible from ob- jects belonging to class BRANCH.	
	(iii)	Write the names of all the members which are accessible from member func- tions of class AUTHOR.	
	(iv)	How many bytes will be required by an object belonging to class AUTHOR?	
3.	(a)	Write a function in C++ to merge the contents of two sorted arrays A & B into third array C. Assuming array A and B are sorted in ascending order and the resultant array C is also required to be in ascending order.	3
	(b)	An array S[40][30] is stored in the memory along the row with each of the ele- ment occupying 2 bytes, find out the memory location for the element S[20][10], if the Base Address of the array is 5000.	3
	(c)	Write a function in C++ to perform Insert operation in a dynamically allocated Queue containing names of students.	4
	(d)	Write a function in C++ to find the sum of both left and right diagonal ele-	2

No.	Questions	Marks
	ments from a two dimensional array (matrix).	
	(e) Evaluate the following postfix notation of expression:	2
	20, 30, +, 50, 40, - ,*	
4.		
(a)	**Observe the program segment given below carefully and fill the blanks marked as Statement 1 and Statement 2 using seekp() and seekg() functions for performing the required task.**	1

```
#include <fstream.h>
class Item
{
int Ino;char Item[20];
public:
//Function to search and display the content from a particular record number void Search(int
);
//Function to modify the content of a particular record number void
Modify(int);
};
void Item::Search(int RecNo)
{
fstream File;
File.open("STOCK.DAT",ios::binary|ios::in);
                                                    //Statement 1
File.read((char*)this,sizeof(Item));
cout<<Ino<<"==>"<<Item<<endl;
File.close();
}
void Item::Modify(int RecNo)
{
fstream File;
File.open("STOCK.DAT",ios::binary|ios::in|ios::out);
```

No.	Questions	Marks

	cout>>Ino;cin.getline(Item,20);
	//Statement 2
	_____
	File.write((char*)this,sizeof(Item));
	File.close();
	}	

(b)	**Write a function in C++ to count the number of lines present in a text file "STORY.TXT".**	2

(c)	**Write a function in C++ to search for a BookNo from a binary file "BOOK.DAT", assuming the binary file is containing the objects of the following class.**
	class
	{
	int Bno;
	char Title[20];
	public:
	int RBno(){return Bno;}
	void Enter(){cin>>Bno;gets(Title);}
	void Display(){cout<<Bno<<Title<<endl;}
	};
		3

5.		
(a)	**What do you understand by Degree and Cardinality of a table?**	2
	Consider the following tables ACTIVITY and COACH and answer (b) and (c) parts of this question:	
	Table: ACTIVITY	

A Code	ActivityName	Stadium	Participants Num	Prize Money	Schedule Date
1001	Relay 100x4	Star Annex	16	10000	23-Jan-2004
1002	High jump	Star Annex	10	12000	12-Dec-2003
1003	Shot Put	Super Power	12	8000	14-Feb-2004
1005	Long Jump	Star Annex	12	9000	01-Jan-2004
1008	Discuss Throw	Super Power	10	15000	19-Mar-2004

No.	Questions	Marks
	Table: COACH	

PCode	Name	Acode	
1	Ahmad Hussain	1001	
2	Ravinder	1008	
3	Janila	1001	
4	Naaz	1003	

(b) **Write SQL commands for the flowing statements:** 4

(i) To display the names of all activities with their Acodes in descending order. (ii) To display sum of PrizeMoney for the Activities played in each of the Stadium separately.

(iii) To display the coach's name and ACodes in ascending order of ACode from the table COACH

(iv) To display the content of the Activity table whose ScheduleDate earlier than 01/01/2004 in ascending order of ParticipantsNum.

(c) **Give the output of the following SQL queries:** 2

(i) SELECT COUNT(DISTINCT ParticipantsNum) FROM ACTIVITY;

(ii) SELECT MAX(ScheduleDate),MIN(ScheduleDate) FROM ACTIVITY; (iii)

```
SELECT Name,ActivityName FROM ACTIVITY A,COACH C
WHERE A.Acode=C.Acode AND A.ParticipantsNum=10; (iv)
SELECT DISTINCT Acode FROM COACH;
```

6.

(a) State and verify Demorgan's Laws algebraically. 2

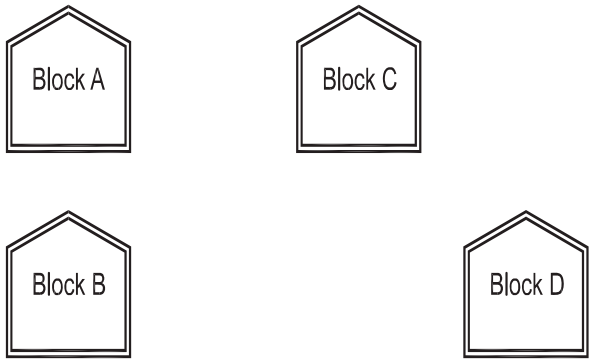
(b) Write the equivalent Boolean Expression for the following Logic Circuit 2

P

Q

No.	Questions	Marks
(c)	Write the POS form of a Boolean function F, which is represented in a truth table as follows:	1

U	V	W	F
0	0	0	1
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

(d)	**Reduce the following Boolean Expression using K-Map:** F(A,B,C,D)= (0,1,2,4,5,6,8,10)	3
7.		
a)	**Compare any two Switching techniques.**	1
b)	**Which of the following is not a Client Side script:** (i) VB Script (ii) Java Script (iii) ASP (iv) PHP	1
c)	**If someone has hacked your Website, to whom you lodge the Complain?**	1
d)	**What do you mean by IP Address? How is it useful in Computer Security? Knowledge**	1
e)	**Supplement Organisation has set up its new center at Mangalore for its office and web based activities. It has 4 blocks of buildings as shown in the diagram below:** 	4

No.	Question	Marks
	Center to center distances between various blocks	

Black A to Block B	50 m
Block B to Block C	150 m
Block C to Block D	25 m
Block A to Block D	170 m
Block B to Block D	125 m
Block A to Block C	90 m

Number of Computers

Black A	25
Block B	50
Block C	125
Block D	10

e1)	Suggest a cable layout of connections between the blocks.	
e2)	Suggest the most suitable place (i.e. block) to house the server of this organisation with a suitable reason.	
e3)	Suggest the placement of the following devices with justification (i) Repeater (ii) Hub/Switch	
e4)	The organization is planning to link its front office situated in the city in a hilly region where cable connection is not feasible, suggest an economic way to connect it with reasonably high speed?	
f)	What do you mean by Spam Mails? How can you protect your mailbox from	1
g)	Spams? Mention any two advantages of Open Source Software over Proprietary Software.	1

COMPUTER SCIENCE (Theory) - Class XII

Solution (Sample Question Paper–III)

Subject Code - 083

TIME : 3 Hrs **MM : 100**

<table>
<tr><th>No.</th><th colspan="2">Answers</th><th>Marks</th></tr>
<tr><td>1.
(a)</td><td colspan="2"></td><td rowspan="4">2</td></tr>
<tr><td></td><td>Global Variable</td><td>Local Variable</td></tr>
<tr><td></td><td>▪ It is a variable which is declared outside all the functions

▪ It is accessible throughout the program</td><td>▪ It is a variable which is declared with in a function or with in a compound statement

▪ It is accessible only within a function/ compound statement in which it is declared</td></tr>
<tr><td></td><td colspan="2"><pre>
#include <iostream.h>
float NUM=900;   //NUM is a global variable
void LOCAL(int T)
{
int Total=0;   //Total is a local variable
for (int I=0;I<T;I++)
Total+=I;
cout<<NUM+Total;
}
void main()
{
LOCAL(45);
}
</pre></td></tr>
<tr><td></td><td colspan="2">(1 Mark for two differences)</td></tr>
</table>

No.	Answers	Marks
	(1 Mark for the suitable example) OR (Full 2 Mark for explanation of differences with the help of an example) OR *(1 Mark for only example with no explanation)*	
(b)	(i) string.h (ii) stdio.h *(½ Mark for mentioning each correct header filename)*	1

(c)	#include <iostream.h>
	class MEMBER
	{
	int Mno;float Fees;
	public:
	void Register(){cin>>Mno>>Fees;}
	void Display(){cout<<Mno<<":"<<Fees<<endl;}
	};
	void main()
	{
	MEMBER M;
	M.Register();
	M.Display();
	}
	(½ Mark each correction)
		2

(d)	111:60 112:70 113:85 *(1 Mark for each correct line of output)*	3

No.	Answers	Marks
(e)	#agaSbarr *(2 Marks for correct line of output)*	2
(f)	(i) ABBC *(2 Marks for mentioning correct option)*	2
2.		

(a)	Data Encapsulation: Wrapping up of data and functions together in a single unit is known as Data Encapsulation. In a class, we wrap up the data and functions together in a single unit.
	Data Hiding: Keeping the data in private visibility mode of the class to prevent it from accidental change is known as Data Hiding.
	class Computer
	{
	char CPU[10];int RAM;
	public: Data Encapsulation
	void STOCK();
	void SHOW();
	};
	(½ Mark each for appropriate definitions)
	(1 Mark for appropriate example showing both)
		2

(b)	i) Destructor, it is invoked as soon as the scope of the object gets over.
	(½ Mark for mentioning destructor)
	(½ Mark for remaining answer)
	ii) Constructor Overloading (or Function Overloading or Polymorphism)
	Seminar S1; //Function 1
	Seminar S2(90); //Function 3
	(½ Mark for mentioning the correct concept)
	(½ Mark for the example)
		2

No.	Answers	Marks

(c)	class TEST
	{
	int TestCode;
	char Description[20];
	int NoCandidate,CenterReqd;
	void CALCNTR();
	public:
	void SCHEDULE();
	void DISPTEST();
	};
	void TEST::CALCNTR()
	{
	CenterReqd=NoCandidate/100 + 1;
	}
	void TEST::SCHEDULE()
	{
	cout<<"Test Code :";cin>>TestCode;
	cout<<"Description :";gets(Description);
	cout<<"Number :";cin>>NoCandidate;
	CALCNTR();
	}
	void TEST::DISPTEST()
	{
	cout<<"Test Code :"<<TestCode<<endl;
	cout<<"Description :"<<Description<<endl;
	cout<<"Number :"<<NoCandidate<<endl;;
	cout<<"Centres :"<<CenterReqd<<endl;;
	}
	(½ Mark for correct syntax for class header)
	(½ Mark for correct declarations of data members)
	(1 Mark for appropriate definition of function CALCNTR())
	(1 Mark for appropriate definition of SCHEDULE() with a call for CALCNTR())
	(1 Mark for appropriate definition of DISPTEST())
		4

(d)	**(i)** None of data members are accessible from objects belonging to class AUTHOR.	4

No.	Answers	Marks
	(1 Mark for correct answer)	
	(ii) Haveit(), Giveit() *(1 Mark for correct answer)*	
	(iii) Data members: Employees, Acode, Aname, Amount Member function: Register(), Enter(), Display(), Haveit(), Giveit(), Start(), Show(), *(1 Mark for correct answer)*	
	(iv) 70 *(1 Mark for correct answer)*	

3.	**(a)** void AddNSave(int A[ ],int B[ ],int C[ ],int N,int M, int &K)
	{
	int I=0,J=0;
	K=0;
	while (I<N && J<M)
	if (A[I]<B[J])
	C[K++]=A[I++];
	else
	if (A[I]>B[J])
	C[K++]=B[J++]
	; else
	{
	C[K++]=A[I++];
	J++;
	}
	for (;I<N;I++)
	C[K++]=A[I];
	for (;J<M;J++)
	C[K++]=B[J];
	}
	(½ Mark for correct Function Header)
	(½ Mark for correct initialization of required variables)
	(½ Mark for correct formation of loop)
	(½ Mark for appropriate conditions and assignments in the loop)
	(½ Mark for appropriately transferring the remaining elements from first array)
	(½ Mark for appropriately transferring the remaining elements from second array)
		3

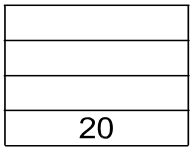
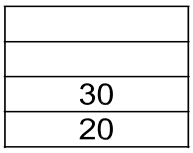
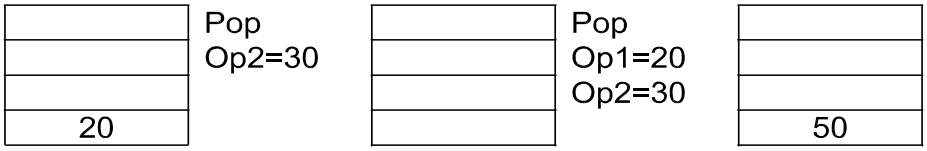
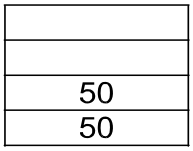
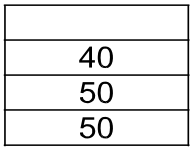
No.	Answers	Marks
(b)	Given, W=2 N=40 M=30 Base(S)=5000 Row Major Formula: Loc(S[I][J]) =Base(S)+W*(M*I+J) Loc(S[20][10]) =5000+2*(30*20+10) =5000+2*(600+10) =5000+1220 =6220 *(1 Mark for writing correct formula (for column major) OR substituting formula with correct values)* *(1 Mark for writing calculation step - at least one step)* *(1 Mark for correct address)*	3

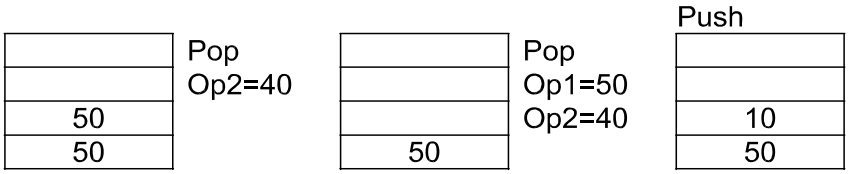
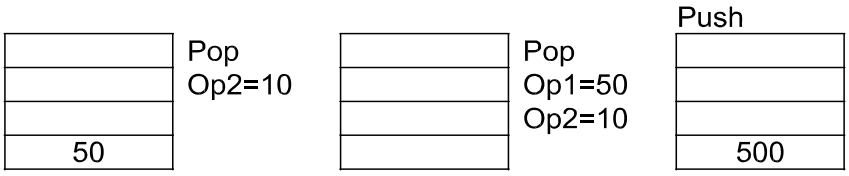
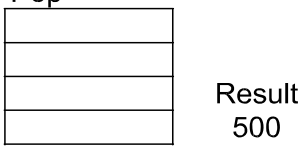
(c)	struct NODE
	{
	char Name[20];
	NODE *Link;
	};
	class QUEUE
	{ NODE *R,*F;
	public:
	QUEUE();
	void Insert();
	void Delete();
	};
	void QUEUE::Insert()
	{
		4

No.	Answers	Marks

	NODE *Temp;
	Temp=new NODE;
	gets(Temp->Name);
	Temp->Link=NULL;
	if (Rear==NULL)
	{
	Rear=Temp;
	Front=Temp;
	}
	else
	{
	Rear->Link=Temp;
	Rear=Temp;
	}
	}
	(1 Mark for creating a new node and assigning/entering appropriate values in it)
	(1 Mark for checking if Queue is Empty)
	(1 Mark for assigning Rear and Front as Temp - if Queue is Empty)
	(1 Mark for eassigning Rear->Link as Front and Rear as Temp)	

(d)	void DiagSum(int M[][4],int N,int M)
	{
	int SumD1=0,SumD2=0;
	for (int I=0;I<N;I++)
	{
	SumD1+=M[I][I];SumD2+=M[N-I-1][I];
	}
	cout<<"Sum of Diagonal 1:"<<SumD1<<endl;
	cout<<"Sum of Diagonal 2:"<<SumD2<<endl;
		2

No.	Answers	Marks
(e)	} *(½ Mark for correct function header)* *(½ Mark for initialization of SumD1 and SumD2 as 0)* *(½ Mark for appropriate loop)* *(½ Mark for correct expression for adding each diagonal elements)* Step 1:Push  Step 2: Push  Step 3: +  Step 4: Push  Step 5: Push  	2

No.	Answers	Marks
	Step 6: -  Step 7:  Step 8: Pop  *(½ Mark for correctly evaluating each operator)* *(½ Mark for the correct result)*	

4.	a)
	File.seekg(RecNo*sizeof(Item)); //Statement 1
	File.seekp(RecNo*sizeof(Item)); //Statement 2
	(½ Mark for each correct Statement)
		1

	(b)
	void CountLine()
	{
	ifstream FIL("STORY.TXT");
	int LINES=0;
	char STR[80];
		2

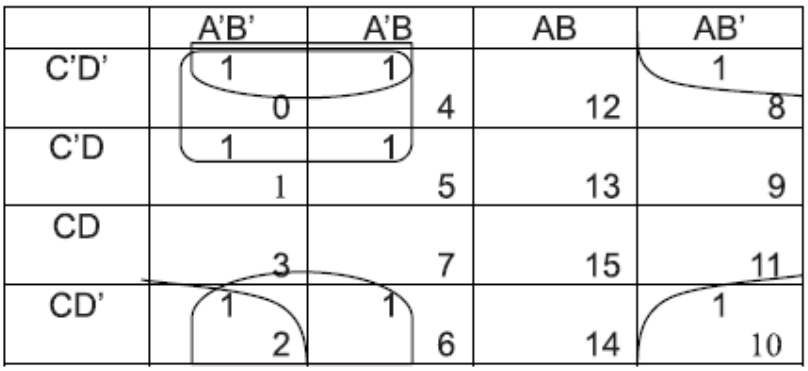
No.	Answers	Marks

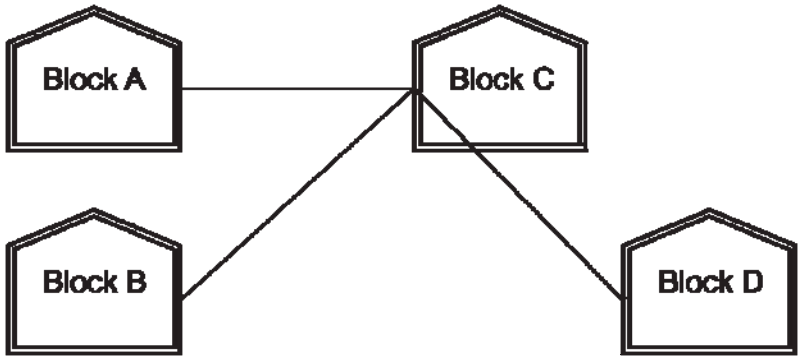
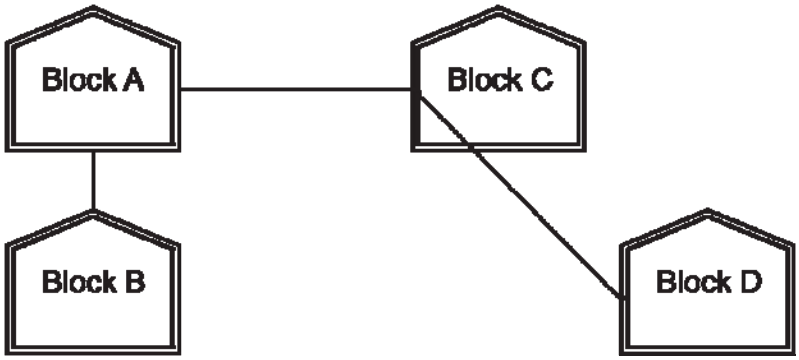
	while (FIL.getline(STR,80))
	LINES++;
	cout<<"No. of Lines:"<<LINES<<endl;
	f.close();
	}
	(½ Mark for opening STORY.TXT correctly)
	(½ Mark for initializing a counter variable as 0)
	(½ Mark for correctly reading a line from the file)
	(½ Mark for correctly incrementing the counter)	

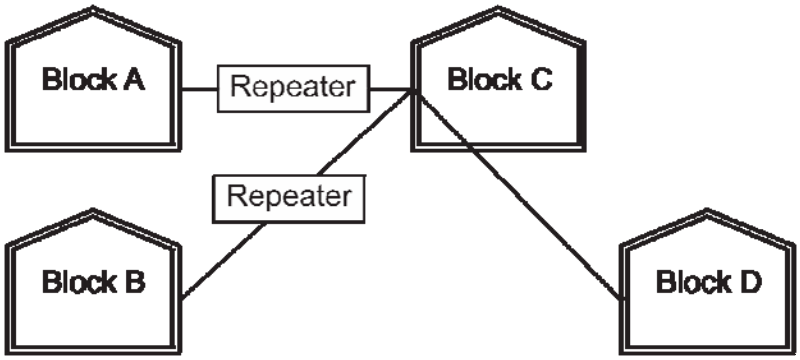
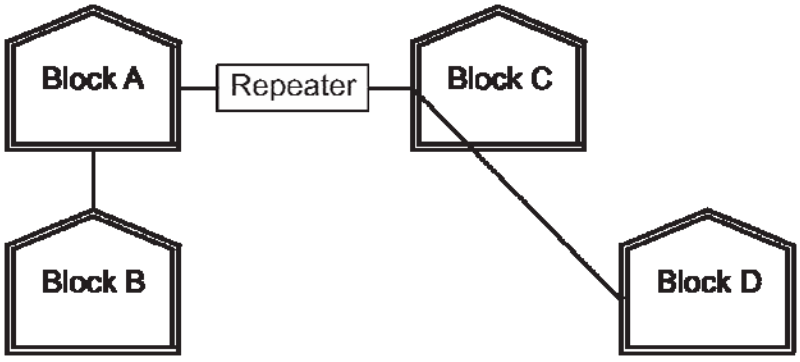
(c)	void BookSearch()
	{
	fstream FIL;
	FIL.open("BOOK.DAT",ios::binary|ios::in);
	BOOK B;
	int bn,Found=0;
	cout<<"Enter Book No. to search…"; cin>>bn;
	while (FIL.read((char*)&S,sizeof(S)))
	if (FIL.RBno()==bn)
	{
	S.Display();
	Found++;
	}
	if (Found==0) cout<<"Sorry! Book not found!!!"<<endl;
	FIL.close();
	}
	(½ Mark for opening BOOK.DAT correctly)
	(½ Mark for reading each record from BOOK.DAT)
	(½ Mark for correct loop / checking end of file)
	(1 Mark for comparing Book number)
	(½ Mark for displaying the matching record)
		3

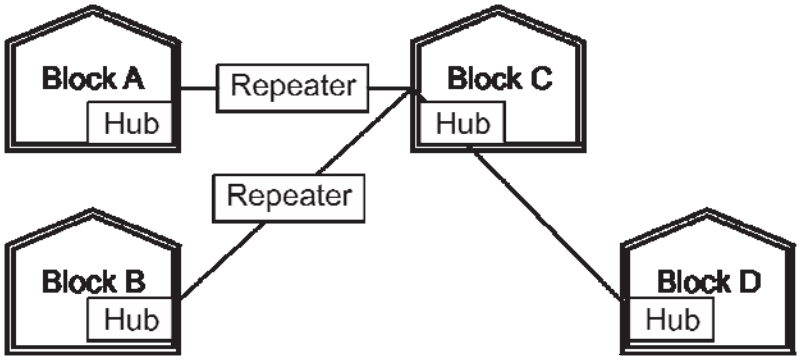
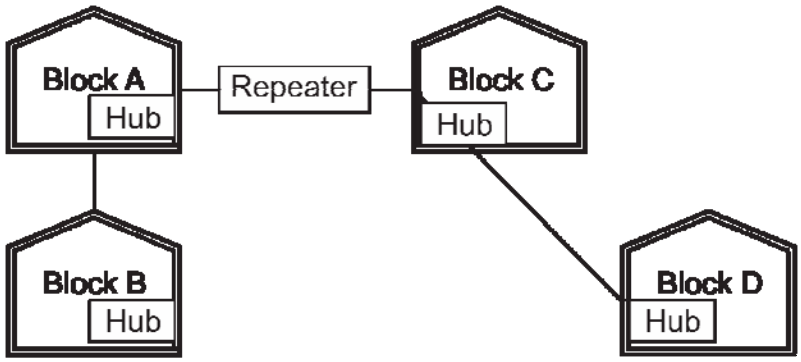
No.	Answers	Marks
5.		
(a)	Degree: Number of Columns in a table Cardinality: Number of rows in a table *(1 Mark for each definition)*	2
(b)	(i) SELECT Acodes, ActivityName FROM ACTIVITY ORDER BY Acode DESC; *(1 Mark for correct query)* OR *(½ Mark for partially correct answer)*	4
	(ii) SELECT SUM(PrizeMoney), Stadium FROM ACTIVITY GROUP BY Stadium; *(1 Mark for correct query)* OR *(½ Mark for partially correct answer)*	
	(iii) SELECT Name, Acode FROM COACH ORDER BY Acode; *(1 Mark for correct query)* OR *(½ Mark for partially correct answer)*	
	(v) SELECT * FROM ACTIVITY WHERE SchduleDate<'01-Jan-2004' ORDER BY ParticipantsNum; *1 Mark for correct query)* OR *(½ Mark for partially correct answer)*	
(c)		2
	(i) 3 *(½ Mark for correct output)*	
	(ii) 19-Mar-2004 12-Dec-2003 *(½ Mark for correct output)*	

No.	Answers	Marks
	(iii) Ravinder Discuss Throw *(½ Mark for correct output)* (iv) 1001 1003 1008 *(½ Mark for correct output)*	
6.	(X+Y)' = X'.Y' Verification (X+Y)'.(X+Y) = X'.Y'.(X+Y) 0 = X'.Y'.X + X'.Y'.Y 0 = X'.X .Y'+ X'.0 0 = 0 .Y'+ 0 0 = 0 + 0 0 = 0 L.H.S = R.H.S *(1 Mark for stating any one of the Demorgan's Law)* *(1 Mark for verifying the law)*	2
(b)	F(P,Q)=(P'+Q).(P+Q') *(2 Marks for the final expression)* OR *(1 Mark for any one of the correct terms out of P'+Q or P+Q')*	**2**
(c)	F(U,V,W) = (U+V+W').(U+V'+W').(U'+V+W') *(1 Mark for the correct expression)*	1

No.	Answers	Marks
(d)		3
	F(A,B,C,D)=A'C'+A'D'+B'D'	
	(½ Mark for placing all 1s at correct positions in K-Map) *(½ Mark for each grouping)* *(1 Mark for writing final expression in reduced/minimal form)* **Note:** Deduct ½ mark if wrong variable names are used	
7.		
a)	Appropriate comparison between any two out of Circuit Switching, Message Switching, Packet Switching (1 Mark for writing Appropriate comparison between any two switching technique)	1
b)	(iii) ASP and (iv) PHP are not client side scripts (1 Mark for correct answer)	1
c)	The complaint has to be lodged with the Police under IT Act (1 Mark for correct answer)	1
d)	An Internet Protocol (IP) address is a numerical identification and logical address that is assigned to devices connected in a computer network. An IP Address is used to uniquely identify devices on the Internet and so one can quickly know the location of the system in the network. *(½ Mark for meaning of IP Address)* *(½ Mark for mentioning the usefulness in network security)*	1

No.	Answers	Marks
e)	**e1)** (Any of the following option) Layout Option 1:  Layout Option 2: Since the distance between Block A and Block B is quite short  *(1 Mark for showing any of the above suitable cable layout)* **e2)** The most suitable place / block to house the server of this □pasm□zation would be Block C, as this block contains the maximum number of computers, thus decreasing the cabling cost for most of the computers as well as increasing the efficiency of the maximum computers in the network. *(½ Mark for suggesting suitable place and ½ for appropriate reason)*	4

No.	Answers	Marks
e3)	(iii) For Layout 1, since the cabling distance between Blocks A and C, and that between B and C are quite large, so a repeater each, would ideally be needed along their path to avoid loss of signals during the course of data flow in these routes.  For layout 2, since the distance between Blocks A and C is large so a repeater would ideally be placed in between this path  *(½ Mark for suggesting suitable place for connecting repeater)*	

No.	Answers	Marks
	(iv) In both the layouts, a hub/switch each would be needed in all the blocks, to interconnect the group of cables from the different computers in each block Layout 1  Layout 2  	
e4)	*(½ Mark for suggesting suitable place for connecting hub)* The most economic way to connect it with a reasonable high speed would be to use radio wave transmission, as they are easy to install, can travel long distances, and penetrate buildings easily, so they are widely used for communication, both indoors and outdoors. Radio waves also have the advantage of being omni directional, which is they can travel in all the directions from the source, so that the transmitter and receiver do not have to be carefully aligned physically.	
f)	*(1 Mark for appropriate answer)* Spam mails, also known as junk e-mail, is a subset of spam that involves nearly identical messages sent to numerous recipients by e-mail. We can protect our mailbox from □pasm by creating appropriate filters. *(½ Mark for the definition of Spam Mails)*	1

No.	Answers	Marks
g)	Open Source's proponents often claim that it offers significant benefits when compared to typical Proprietary Software. Proprietary Software typically favour visible features (giving marketing advantage) over harder-to measure qualities such as stability, security and similar less glamorous attributes. Open Source Software developers are evidently motivated by many factors but favouring features over quality is not noticeable amongst them. For many developers, peer review and acclaim is important, so it's likely that they will prefer to build software that is admired by their peers. Highly prized factors are clean design, reliability and maintainability, with adherence to standards and shared community values preeminent. *(1 Mark for appropriate answer)*	1

TIPS FOR THE EXAM AND TIME MAMANGEMENT

In c++ execution always starts from main() function, so for solving output questions however big the code may be always start from main() .Create(draw) memory locations in a page(usually the last page in the answer book) and write the values systematically with pencil. Do not leave any question unanswered because each steps carry marks. Here are some general tips for the exam please go through it.

***LOOK AFTER YOURSELF*:**

Diet : Try and eat a healthy diet, not just chips and burgers! Take time out for meals; Don't try to study while you're eating .

Rest : Don't try to study through the late night before an exam .Go to bed in good time and get minimum sleep of 5 to 6 hours a day.

TRUST YOURSELF:

Trust your memory and prepare with care.

Once you know a subject/Topic thoroughly, move on to the next.

Plan your study judiciously well and if possible study with your friends.

Prepare a proper timetable for study and follow it strictly, feel that you are your master.

PACE YOURSELF AT MAXIMUM OF YOUR LIMIT:

Use your time well.

Do not try to do too much at once.

Take a break from time to time in each study period. (ONE HOUR IN ONE SEATING)

USING OLD EXAM PAPERS AND OLD RESOURCES :

Look through old exam papers.

Make outline plans for the answers.

Note carefully the slight differences in how questions are asked from year to year.

REVISING FOR EXAMS:

Here are FEW tips on revising for exams:

Make use of your OWN learning style when you revise & learn.

TAKING EXAMS:

HERE ARE EIGHT TIPS FOR HELPING YOU TO COPE ON THE DAY OF AN EXAM

Carry the original hall ticket to the examination hall, keeping one Xerox copy of the hall ticket at Home.

- **Write the hall ticket number on the main answer sheet seeing the hall ticket and also on every additional sheet taken.**
- **Write the total number of additional sheets taken on the main answer sheet.**
- **Write the page number on the additional sheet.**

REVIEW COMMITTEE MEMBERS

Advisor : **Dr. E. T Arasu**
Director
ZIET Mysore

Guidance : **Mrs. Ruby Huria**
Asst Educational Offficer
ZIET Mysore

Subject Contributor :

Mr. Jitendra Sharma
PGT Computer Science
KV CRPF Yelahanka

Mrs. Leena A
PGT Computer Science
KV Mysore