

LAB MANUAL
Object Oriented Programming Concepts Lab
(ITX-221)

DEPARTMENT OF INFORMATION TECHNOLOGY



**Dr. B R AMBEDKAR NATIONAL INSTITUTE OF TECHNOLOGY
JALANDHAR – 144011**

Table of Contents

S. No.	Name of Practicals	Page No.
1.	Write a program to read a matrix of size m x n from the keyboard and display the same using function.	3
2.	Program to make the use of inline function.	5
3.	Write a function power () which raise a number m to a power n. The function takes double value of m and integer value of n and returns the result. Use a default value of n is 2 to make the function to calculate squares when this argument is omitted.	6
4.	Write a class STRING that can be used to store strings, add strings, equate string, output strings.	8
5.	Create the class TIME to store time in hours and minutes. Write a friend function to add two TIME objects.	10
6.	Create two classes DM and DB. DM stores the distance in meter and centimeters and DB stores the distance in feet and inches. Write a program to add object of DM with the object of DB class.	12
7.	a) Program to demonstrate the concept of Default Constructor. b) Program to demonstrate the concept of Parameterized Constructor. c) Program to demonstrate the concept of Copy Constructor. d) Program to demonstrate the concept of Constructor Overloading.	14
8.	Program to demonstrate the concept of Destructor.	19
9.	Program to show Multiple Inheritance.	21
10.	Program to show Multilevel Inheritance.	23
11.	Program to show Hybrid Inheritance.	26
12.	Program to overload Unary operator.	29
13.	Program to overload Binary operator.	31
14.	Program to read the Name and Roll Number of Students from keyboard and write them into a file and then display it.	33
15.	Program to copy one file onto the end of another, adding line numbers.	35
16.	Write a Function Template for finding the minimum value contained in an	37

	array.	
--	--------	--

1. Write a program to read a matrix of size $m \times n$ from the keyboard and display the same using function.

PROGRAM:-

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

void matrix(int m, int n)
{
int a[20][20],i,j;

cout<<"\nEnter Matrix :\n";
for(i=1;i<=m;i++)
{
for(j=1;j<=n;j++)
{
cin>>a[i][j];
}
}

cout<<"\n\nMatrix is :\n\n";
for(i=1;i<=m;i++)
{
for(j=1;j<=n;j++)
{
cout<<a[i][j]<<"\t";
}
cout<<endl;
}
}

void main()
{
clrscr();
int m,n;

cout<<"Enter No. of Rows : ";
cin>>m;
cout<<"Enter No. of Columns : ";
cin>>n;

matrix(m,n);
getch();
}
```

OUTPUT:-

Enter No. of Rows : 2

Enter No. of Columns : 3

Enter Matrix :

3

4

11

6

8

3

Matrix is :

3	4	11
6	8	3

2. Program to make the use of inline function.

PROGRAM:-

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



```

inline int Square(int a)
{
return a*a;
}

inline int Cube(int b)
{
return b*b*b;
}

void main()
{
clrscr();
int a,b;

cout<<"Enter any Number : ";
cin>>a;
int s=Square(a);
cout<<"\nSquare of Number is : "<<s;

cout<<"\n\n\nEnter any Number : ";
cin>>b;
int c=Cube(b);
cout<<"\nCube of Number is : "<<c;

getch();
}

```

OUTPUT:-

Enter any Number : 5
 Square of Number is : 25

Enter any Number : 3
 Cube of Number is : 27

3. Write a function power () which raise a number m to a power n. The function takes double value of m and integer value of n and returns the result. Use a default value of n is 2 to make the function to calculate squares when this argument is omitted.

PROGRAM:-

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



```
#include<conio.h>

float power(float m, int n=2)
{
    float total=1;
    for(int i=1;i<=n;i++)
    {
        total=total*m;
    }
    return total;
}

void main()
{
    clrscr();
    float m,r;
    int n;
    cout<<"Enter value of m : ";
    cin>>m;
    cout<<"Enter value of n : ";
    cin>>n;

    r=power(m,n);
    cout<<"\nResult : "<<r;

    r=power(m);
    cout<<"\nResult (n is omitted) : "<<r;

    getch();
}
```

OUTPUT:-

Enter value of m : 2.5

Enter value of n : 3

Result : 15.625

Result (n is omitted) : 6.25



4. Write a class STRING that can be used to store strings, add strings, equate string, output strings.

PROGRAM:-

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



```
class STRING
{
char a[50],b[50];

public:
void Input()
{
cout<<"Enter First String : ";
cin.getline(a,50);
cout<<"Enter Second String : ";
cin.getline(b,50);
}

void Output()
{
cout<<"\nEntered Strings are :";
cout<<endl<<a;
cout<<endl<<b;
}

void Equate()
{
int c=strcmp(a,b);
if(c==0)
cout<<"\n\nStrings are same.";
else
cout<<"\n\nStrings are not same.";
}

void Add()
{
cout<<"\n\nString after Concatenation :\n";
cout<<strcat(a,b);
}
};

void main()
{
clrscr();
STRING obj;
obj.Input();
obj.Output();
obj.Equate();
obj.Add();
}
```



```
getch();  
}
```

OUTPUT:-

Enter First String : NIT Jalandhar
Enter Second String : Dept. Information Technology

Entered Strings are :
NIT Jalandhar
Dept. Information Technology

Strings are not same.

Strings after Concatenation :
NIT Jalandhar Dept. Information Technology

5. Create the class TIME to store time in hours and minutes. Write a friend function to add two TIME objects.

PROGRAM:-

```
#include<iostream.h>  
#include<conio.h>  
#include<string.h>  
  
class TIME  
{
```



```

int hours,minutes,tmin,hrs,min;

public:
void getTime()
{
cout<<"Enter Hours : ";
cin>>hours;
cout<<"Enter Minutes : ";
cin>>minutes;
}

void addTime(TIME t1, TIME t2)
{
tmin=((t1.hours*60)+t1.minutes)+((t2.hours*60)+t2.minutes);
hrs=(tmin/60);
min=(tmin%60);
cout<<"\nTotal Time is : "<<hrs<<" Hours and "<<min<<" Minutes";
}
};

void main()
{
clrscr();
TIME t1,t2,t3;

cout<<"Enter First Time\n";
t1.getTime();
cout<<"\n\nEnter Second Time\n";
t2.getTime();

t3.addTime(t1,t2);
getch();
}
OUTPUT:-

```

Enter First Time
 Enter Hours : 6
 Enter Minutes : 34

Enter Second Time
 Enter Hours : 7
 Enter Minutes : 42

Total Time is : 14 Hours and 16 Minutes



6. Create two classes DM and DB. DM stores the distance in meter and centimeters and DB stores the distance in feet and inches. Write a program to add object of DM with the object of DB class.

PROGRAM:-

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

```
class DM
{
public:
int meter,cm;

public:
void Distance1()
```



```

{
cout<<"Enter Meters : ";
cin>>meter;
cout<<"Enter Centimeters : ";
cin>>cm;
}
};

class DB
{
public:
int feet,inch;

public:
void Distance2()
{
cout<<"Enter Feets : ";
cin>>feet;
cout<<"Enter Inches : ";
cin>>inch;
}
};

void main()
{
clrscr();
DM ob1;
DB ob2;
int centimeters,inches,tcm,tinch,totalm,totalcm,totalfeet,totalinch;
cout<<"\nEnter First Distance\n";
ob1.Distance1();
cout<<"\nEnter Second Distance\n";
ob2.Distance2();

centimeters=(ob1.meter*100)+ob1.cm;
inches=(ob2.feet*12)+ob2.inch;

tcm=centimeters+(inches*2.54);
totalm=(tcm/100);
totalcm=(tcm%100);

cout<<"\n\nTotal Distance in Meters and centimeters is :\n\n";
cout<<totalm<<" Meters and "<<totalcm<<" centimeters";
}

```



```

tinch=(centimeters/2.54)+inches;
totalfeet=(tinch/12);
totalinch=(tinch%12);

cout<<"\n\n\nTotal Distance in Feets and Inches is :\n\n";
cout<<totalfeet<<" Feets and "<<totalinch<<" Inches";

getch();
}

```

OUTPUT:-

Enter First Distance
 Enter Meters : 2
 Enter Centimeters : 76

Enter Second Distance
 Enter Feets : 13
 Enter Inches : 7

Total Distance in Meters and centimeters is :
 6 Meters and 90 centimeters

Total Distance in Feets and Inches is :
 22 Feets and 7 Inches

7. a) Program to demonstrate the concept of Default Constructor.

PROGRAM:-

```

#include<iostream.h>
#include<conio.h>

class Example
{
int a,b;

public:
Example()
{
a=10;
b=20;
}

```



```
void Display()
{
    cout<<"Values are :\n"<<a<<endl<<b;
}
};

void main()
{
    clrscr();

    Example obj;
    obj.Display();

    getch();
}
```

OUTPUT:-

Values are :
10
20

7. b) Program to demonstrate the concept of Parameterized Constructor.**PROGRAM:-**

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

class Example
{
    int a,b;

public:
    Example(int x,int y)
    {
        a=x;
        b=y;
    }
}
```



```

void Display()
{
cout<<"Values are :\n"<<a<<endl<<b;
}
};

void main()
{
clrscr();
Example obj(20,40);
obj.Display();

getch();
}

```

OUTPUT:-

Values are :
20
40

7. **c) Program to demonstrate the concept of Copy Constructor.**

PROGRAM:-

```

#include<iostream.h>
#include<conio.h>
class Example
{
    int a,b;

public:
    Example(int x,int y)
    {
        a=x;
        b=y;
    }
    void Display()
    {
        cout<<"\nValues :"<<a<<"\t"<<b;
    }
};

void main()
{

```



```
clrscr();
Example Object(15,30);
Example Object2=Object;

Object.Display();
Object2.Display();

getch();
}
```

OUTPUT:-

```
Values : 15    30
Values : 15    30
```

7. d) Program to demonstrate the concept of Constructor Overloading.**PROGRAM:-**

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

class Example
{
int a,b;

public:
Example()
{
a=10;
b=20;
}

Example(int x,int y)
{
a=x;
b=y;
}
```

```
void Display()
```



```
{  
cout<<"\nValues are :\n"<<a<<endl<<b;  
}  
};
```

```
void main()  
{  
clrscr();  
Example obj;  
Example obj2(30,60);  
obj.Display();  
obj2.Display();  
getch();  
}
```

OUTPUT:-

Values are :

10

20

Values are :

30

60



8. Program to demonstrate the concept of Destructor.

PROGRAM:-

```
#include<iostream.h>
#include<conio.h>
#include<string.h>
class Student
{
private:
char Name[25];
int Roll_No;

public:
Student()
{
strcpy(Name,"Vikas");
Roll_No=121;
}
void Display()
{
cout<<"\nName : "<<Name;
cout<<"\nRoll No. : "<<Roll_No;
}
~Student()
{
cout<<"\nDestroying Object...";
}
};

void main()
{
Student obj;
obj.Display();
getch();
}
```



OUTPUT:-

Name : Vikas
Roll No. : 121
Destroying Object...

9. Program to show Multiple Inheritance.



PROGRAM:-

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

class Student
{
protected:
int Roll_No;
int Marks1,Marks2;

public:
void getData()
{
cout<<"Enter Roll No. : ";
cin>>Roll_No;
cout<<"\nEnter Marks in Subject 1 : ";
cin>>Marks1;
cout<<"\nEnter Marks in Subject 2 : ";
cin>>Marks2;
}
};

class Sports
{
protected:
int S_Marks;

public:
void getMarks()
{
cout<<"\nEnter Marks in Sports : ";
cin>>S_Marks;
}
};

class Result : public Student, public Sports
{
int total,per;

public:
void showResult()
{
total=(Marks1+Marks2+S_Marks);
cout<<"\n\nTotal Marks is : "<<total<<" out of 300"<<endl;
}
};
```



```
per=(total*100)/300;
cout<<"\nPercentage is : "<<per;
}
};

void main()
{
clrscr();
Result obj;
obj.getData();
obj.getMarks();
obj.showResult();

getch();
}
```

OUTPUT:-

Enter Roll No. : 10283

Enter Marks in Subject 1 : 76
Enter Marks in Subject 2 : 57
Enter Marks in Sports : 85

Total Marks is : 218 out of 300

Percentage is : 72

10. Program to show Multilevel Inheritance.**PROGRAM:-**

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



```

class Student
{
protected:
int Roll_No;
char Name[50];

public:
void getData()
{
cout<<"\nEnter Name of Student : ";
cin.getline(Name,50);
cout<<"\nEnter Roll No. of Student : ";
cin>>Roll_No;
}

public:
void showData()
{
cout<<"\n\nStudent Name is : "<<Name;
cout<<"\n\nRoll No. is : "<<Roll_No;
}
};

class Marks : public Student
{
protected:
int Marks1,Marks2;

public:
void getMarks()
{
cout<<"\nEnter Marks of "<<Name<<" in Subject 1 : ";
cin>>Marks1;
cout<<"\nEnter Marks of "<<Name<<" in Subject 2 : ";
cin>>Marks2;
}
};

class Result : public Marks
{
int total,per;
public:
void showResult()
{

```



```
total=(Marks1+Marks2);
cout<<"\n\nTotal Marks is : "<<total<<" out of 200"<<endl;

per=(total*100)/200;
cout<<"\nPercentage is : "<<per;

if(per>=60)
cout<<"\n\nFirst Division";
else if(per>=50)
cout<<"\n\nSecond Division";
else
cout<<"\n\nThird Division";
}

};

void main()
{
clrscr();
Result obj;
obj.getData();
obj.getMarks();
obj.showData();
obj.showResult();

getch();
}
```

OUTPUT:-

Enter Name of Student : Vikas
Enter Roll No. of Student : 10283

Enter Marks of Vikas in Subject 1 : 76
Enter Marks of Vikas in Subject 1 : 57



Student Name is : Vikas
Roll No. is : 10283

Total Marks is 133 out of 200
Percentage is : 66
First Division

11. Program to show Hybrid Inheritance.

PROGRAM:-

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

```
class Arithmetic
{
protected:
int a,b;
```



```

public:
void getData()
{
cout<<"\nEnter First Number : ";
cin>>a;
cout<<"\nEnter Second Number : ";
cin>>b;
}
};

class Add: public Arithmetic
{
protected:
int sum;

public:
void Addition()
{
cout<<"\n\n---- Addition ----\n";
getData();
sum=a+b;
}
};

class Minus: public Arithmetic
{
protected:
int sub;

public:
void Subtract()
{
cout<<"\n\n---- Subtraction ----\n";
getData();
sub=a-b;
}
};

class Result: public Add, public Minus
{
public:
void DisSum()
{
cout<<"\nSum is : "<<sum<<endl<<endl;
}
};

```



```
void DisSub()
{
    cout<<"\nDifference is : "<<sub<<endl<<endl;
}
};

void main()
{
    clrscr();

    Result obj;
    obj.Addition();
    obj.DisSum();
    obj.Subtract();
    obj.DisSub();

    getch();
}
```

OUTPUT:-

---- Addition ----

Enter First Number : 34
Enter Second Number : 60
Sum is : 94

---- Subtraction ----

Enter First Number : 27
Enter Second Number : 18
Difference is : 9



12. Program to overload Unary operator.

PROGRAM:-

```
#include<iostream.h>
#include<conio.h>
class s
{
int a,b,c;
public:
void getdata();
void display();
void operator-();
};

void s::getdata()
{
cout<<"Enter Values :\n";
}
```



```
    cin>>a;
    cin>>b;
    cin>>c;
}
void s::display()
{
    cout<<"\nOutput :\n";
    cout<<a<<"\t";
    cout<<b<<"\t";
    cout<<c<<endl;
}
void s::operator-()
{
    a=-a;
    b=-b;
    c=-c;
}
```

```
void main()
```

```
{
clrscr();
s a1;
a1.getdata();
-a1;
a1.display();
getch();
}
```

OUTPUT:-

Enter Values :

23

-2

45

Output :

-23 2 -45



13. Program to overload Binary operator.

PROGRAM:-

```
#include<iostream.h>
#include<conio.h>
class num
{
private:
int a,b,c;
public:
void input();
void show();
num operator+(num);
};

void num ::input()
{
cout<<"Enter the values for a,b,c :\n";
cin>>a>>b>>c;
}

void num::show()
```



```

{
cout<<"\tA=<<a<<"\tB=<<b<<"\tC=<<c<<"\n";
}

num num::operator+(num t)
{
num temp;
temp.a=a+t.a;
temp.b=b+t.b;
temp.c=c+t.c;
return(temp);
}

void main()
{
clrscr();
num x,y,z;
cout<<"\nObject x\n";
x.input();
cout<<"\nObject y\n";
y.input();

z=x+y;
cout<<"\nx:";
x.show();
cout<<"y:";
y.show();
cout<<"\nz:";
z.show();
getch();
}

```

OUTPUT:-

Object x
Enter the values for a,b,c :
23
12
18

Object y
Enter the values for a,b,c :
45
4



31

```
x:      A=23  B=12  C=18
y:      A=45  B=4   C=31
z:      A=68  B=16  C=49
```

- 14. Program to read the Name and Roll Number of Students from keyboard and write them into a file and then display it.**

PROGRAM:-

```
#include<iostream.h>
#include<fstream.h>
#include<conio.h>
void main()
{
clrscr();
char Name[50];
int RollNo;

ofstream outFile;
outFile.open("Students.txt");

cout<<"\nEnter the Name of Student : ";
cin>>Name;
outFile<<Name<<endl;
cout<<"\nEnter the Roll No. of Student : ";
cin>>RollNo;
outFile<<RollNo;

outFile.close();

cout<<"\n\nData is Saved to File";

char data[50];
ifstream inFile;
inFile.open("Students.txt");

inFile>>data;
cout<<"\n\nStudent Name : "<<data;
```



```
inFile>>data;  
cout<<"\n\nRoll No. : "<<data;  
  
inFile.close();  
  
getch();  
}
```

OUTPUT:-

Enter the Name of Student : Vikas
Enter the Roll No. of Student : 233

Data is Saved to File

Student Name : Vikas
Roll No. : 233



15. Program to copy one file onto the end of another, adding line numbers.**PROGRAM:-**

```
#include<iostream.h>
#include<fstream.h>
#include<conio.h>
void main()
{
clrscr();
char data[50];
int LineNo=1;

ifstream File1;
ofstream File2;
File1.open("F1.txt");
File2.open("F2.txt",ios::app);

while(!File1.eof())
{
File1>>data;
File2<<LineNo<<. " <<data<<endl;
LineNo++;
}

File1.close();
File2.close();

cout<<"Press any Key to Continue... ";
getch();
}
```



OUTPUT:-

F1.txt

Monitor
Keyboard
Mouse
CPU
UPS

F2.txt

1. Monitor
2. Keyboard
3. Mouse
4. CPU
5. UPS



16. Write a Function Template for finding the minimum value contained in an array.

PROGRAM:-

```
#include <iostream.h>
#include <conio.h>
template <class T>
T findMin(T arr[],int n)
{
    int i;
    T min;
    min=arr[0];
    for(i=0;i<n;i++)
    {
        if(min > arr[i])
            min=arr[i];
    }
    return(min);
}

void main()
{
    clrscr();
    int iarr[5];
    char carr[5];
    double darr[5];

    cout<<"\nEnter Integer Values \n";
    for(int i=0;i<5;i++)
    {
        cout<<"Value "<<i+1<<" : ";
        cin>>iarr[i];
    }
    cout<<"\nEnter Character Values \n";
    for(int j=0;j<5;j++)
    {
        cout<<"Value "<<j+1<<" : ";
        cin>>carr[j];
    }

    cout<<"\nEnter Decimal Values \n";
    for(int k=0;k<5;k++)
    {
```



```
cout<<"Value "<<k+1<<" : ";
cin>>darr[k];
}

cout<<"Integer Minimum is : "<<findMin(iarr,5)<<"\n";
cout<<"Character Minimum is : "<<findMin(carr,5)<<"\n";
cout<<"Decimal Minimum is : "<<findMin(darr,5)<<"\n";
getch();
}
```

OUTPUT:-

```
Enter Integer Values
Value 1 : 12
Value 2 : 3
Value 3 : 56
Value 4 : 2
Value 5 : 45
```



Enter Character Values

Value 1 : d

Value 2 : f

Value 3 : c

Value 4 : h

Value 5 : g

Enter Decimal Values

Value 1 : 2.4

Value 2 : 3.12

Value 3 : 34

Value 4 : 1.19

Value 5 : 8.0

Integer Minimum is : 2

Character Minimum is : c

Decimal Minimum is : 1.19

