EXPERIMENT NUMBER –Practical 7.1

STUDENT'S NAME –

STUDENT'S UID –

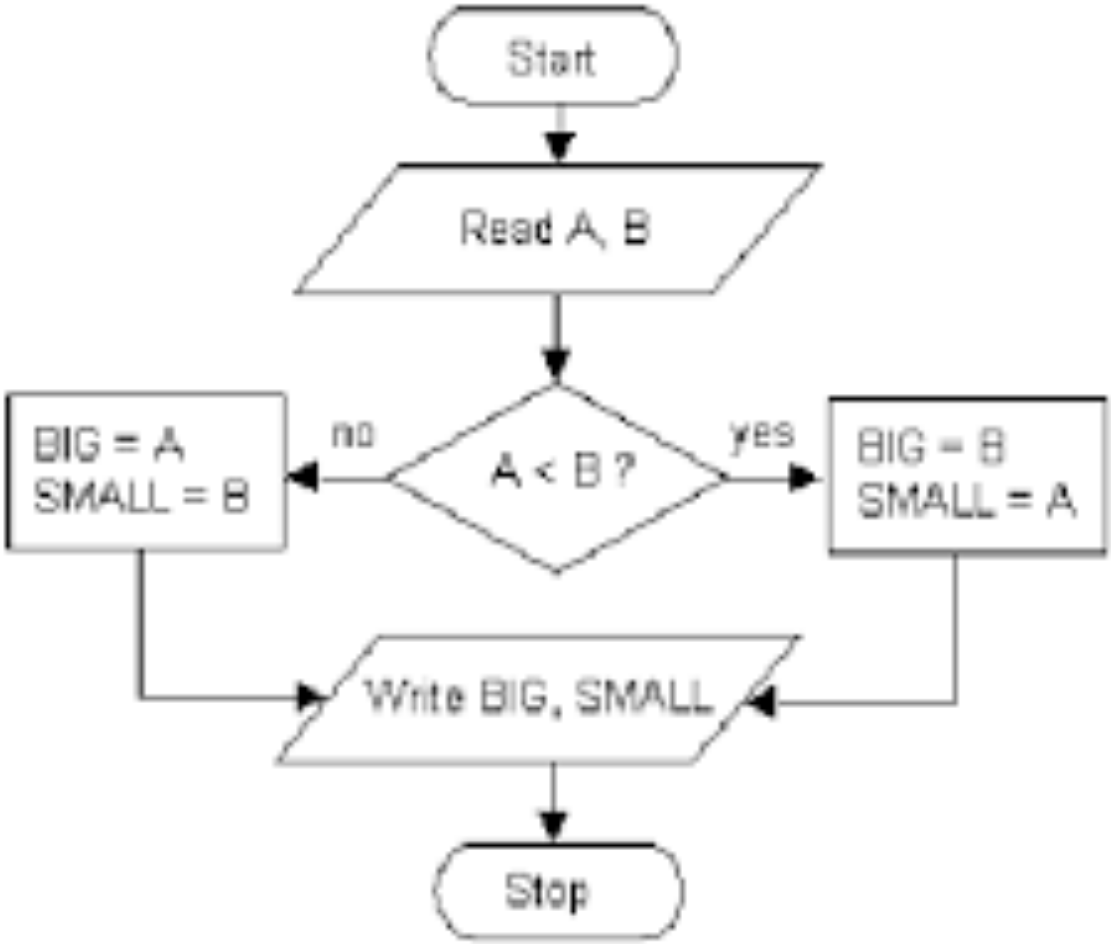
CLASS AND GROUP –

SEMESTER –2ND

TOPIC OF EXPERIMENT –Programs based on Run-time polymorphism.

AIM OF THE EXPERIMENT- WAP to create a class that will maintain the records of person with details (Name and Age) and find the eldest among them. The program must use this pointer to return the result by overloading> operator among two objects.

FLOWCHART/ ALGORITHM -



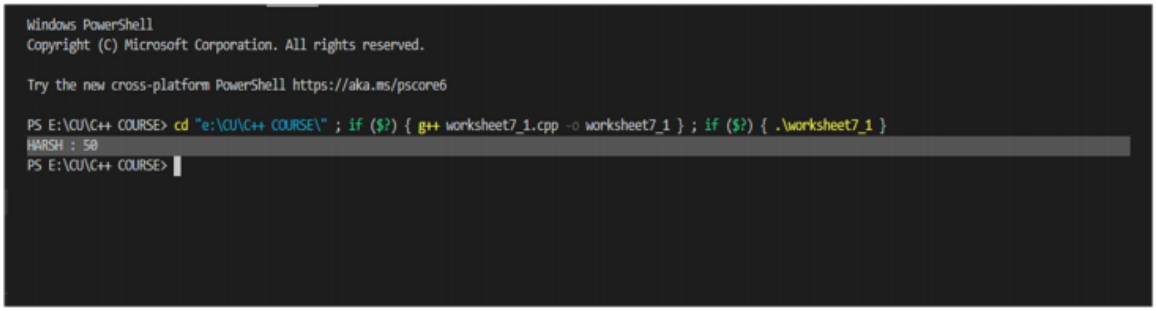
PROGRAM CODE-

```
#include<iostream> using
namespace std; class
Records
{
    int age; string
    name; public:
        Records() {};
        Records(string n,int a): name(n),age(a)
        {

        }
        void show()
        {
            cout<<name<<" : "<<age<<endl;
        }
        Records eldest(Records o)
        {
            return (o.age>age)? o: *this;
        }
};
int main()
{
    Records ob[3]={Records("SINGH",21),Records("HARSH",50),Records("Ram",30)};
    Records res;
    for(int i=0;i<2;i++)
        res = ob[i].eldest(ob[i+1]); res.show();
    return 0;
}
```

PROGRAMS' EXPLANATION (in brief)-

In the above program class is created which maintain the record of person such as name age and will find the eldest one using pointer function and the result is given by overloading operator among two object.

OUTPUT-



EXPERIMENT NUMBER –Practical 7.2

STUDENT'S NAME – YASH RAJ

STUDENT'S UID – 21BCS11765

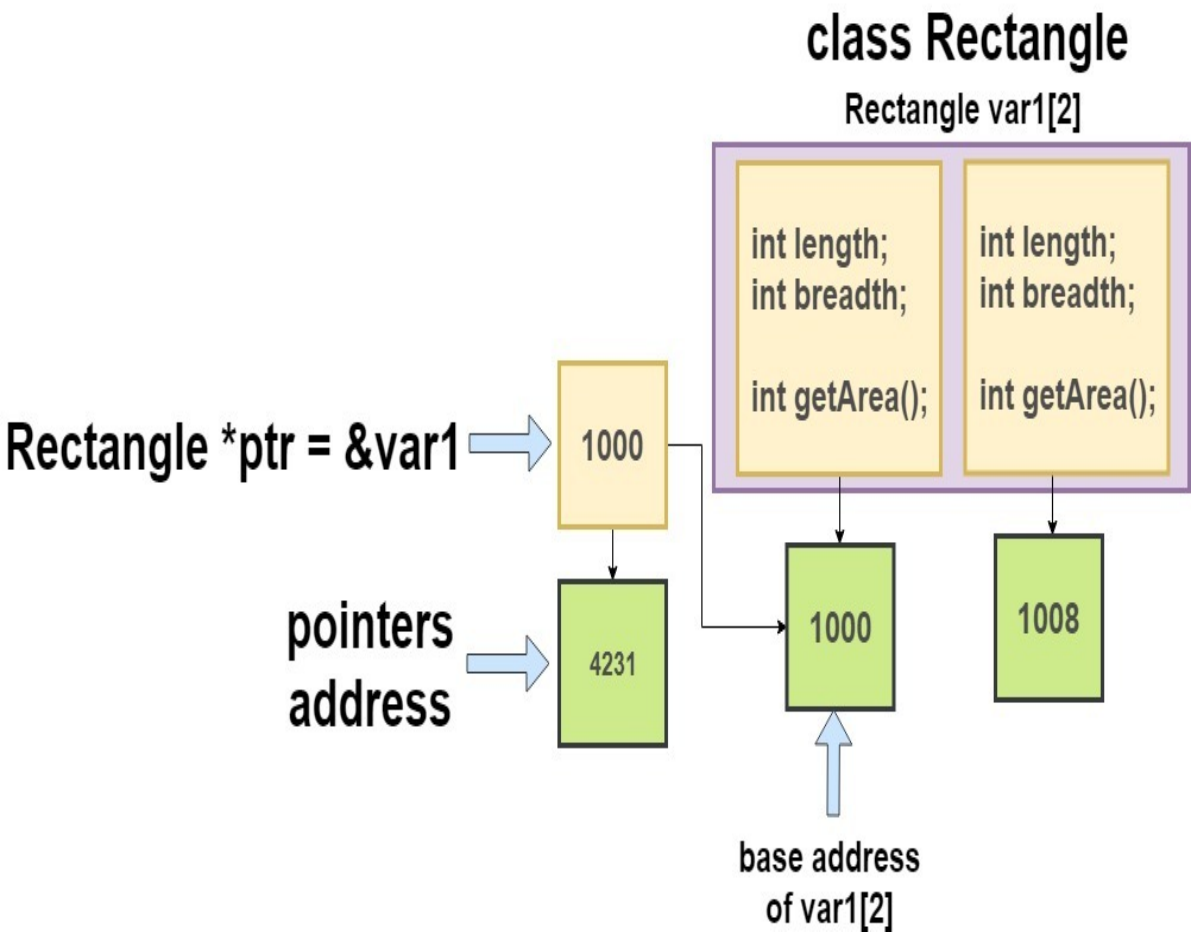
CLASS AND GROUP – 509-B

SEMESTER – 2ND

TOPIC OF EXPERIMENT – **Programs based on Run-time polymorphism.**

AIM OF THE EXPERIMENT - **WAP TO ACCESS MEMBER USING POINTER TO OBJECT MEMBER.**

FLOWCHART/ ALGORITHM -



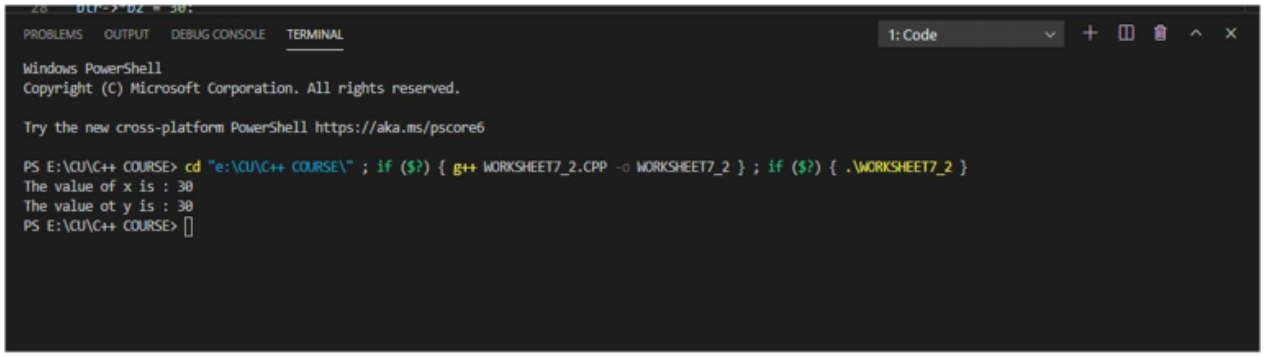
PROGRAM CODE-

```
#include<iostream>
using namespace std;
class A
{
public:
int x,y;
};
int main()
{
A ob;

//Pointer to object
A *ptr = &ob;
int A :: *p1 = &A :: x;
int A :: *p2 = &A :: y;

//Using pointer to object to access data member, pointed by a pointer
ptr->*p1 = 30;

//Using pointer to object to access data member, pointed by a pointer
ptr->*p2 = 30;

cout<<"The value of x is : " << ptr->*p1 << "\n";
cout<<"The value of y is : " << ptr->*p2 << "\n";
}
```

PROGRAMS' EXPLANATION (in brief)-

In the above program two pointer object is created and by using pointer to object function the data member are accessed which is pointed by the pointer.

OUTPUT-



EXPERIMENT NUMBER –Practical 7.3

STUDENT'S NAME – YASH RAJ

STUDENT'S UID - 21BCS11765

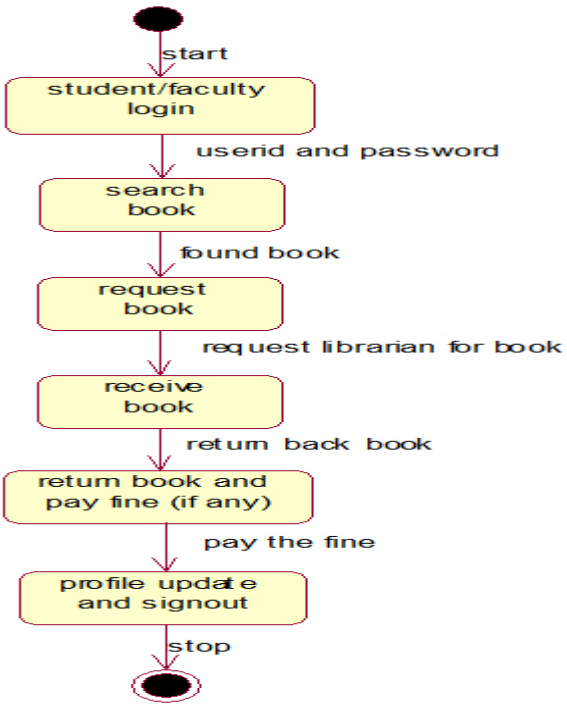
CLASS AND GROUP – 509-B

SEMESTER – 2ND

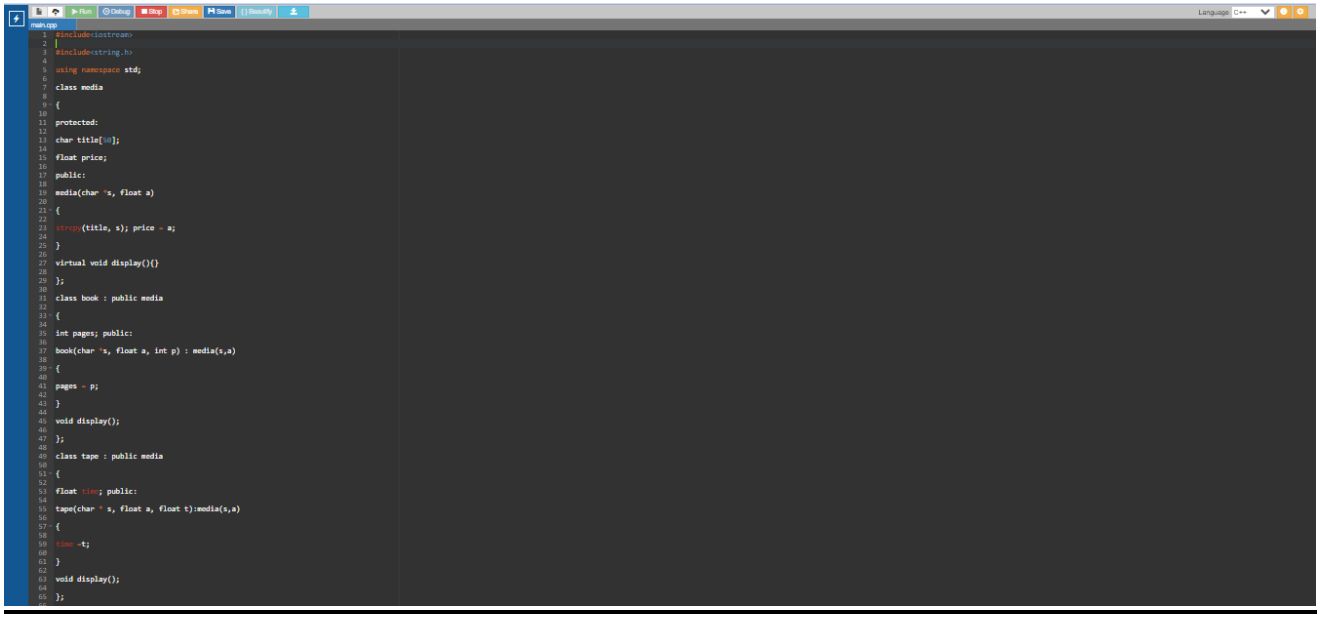
TOPIC OF EXPERIMENT – Programs based on Run-time polymorphism.

AIM OF THE EXPERIMENT - WAP to design a class representing the information regarding digital library (books, tape: book & tape should be separate classes having the base class as media).The class should have the functionality for adding new item, issuing, deposit etc. The program should link the objects with concerned function by the concept of runtime polymorphism.

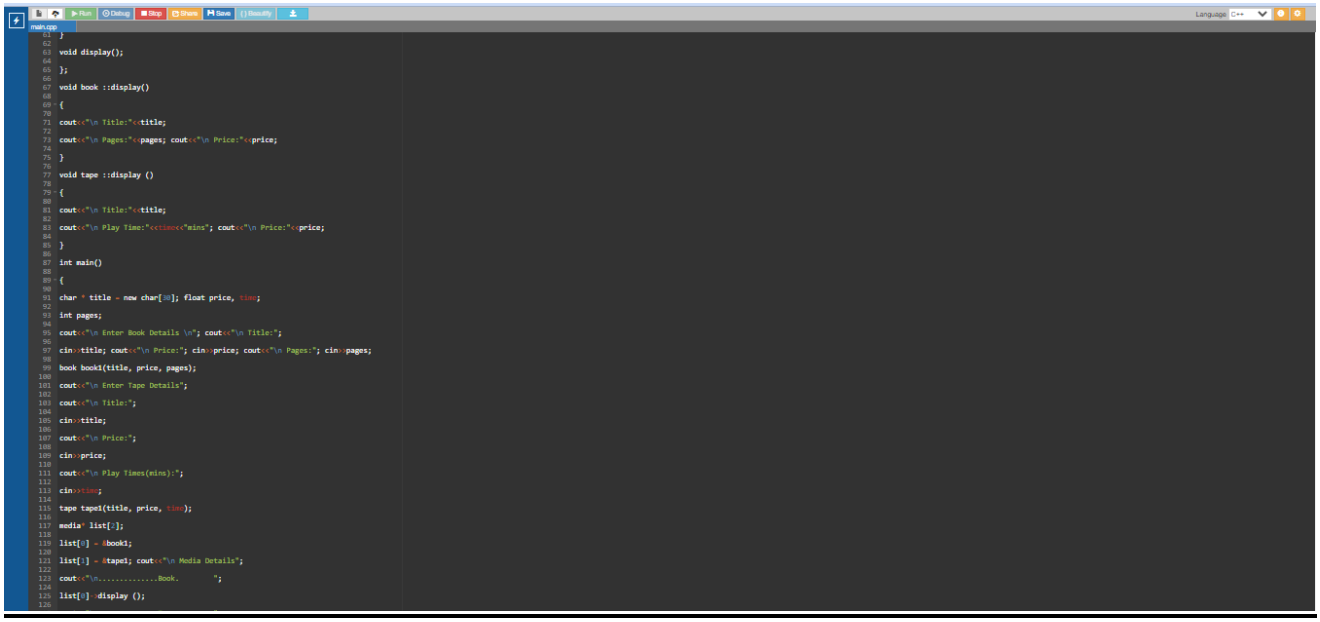
FLOWCHART/ ALGORITHM-



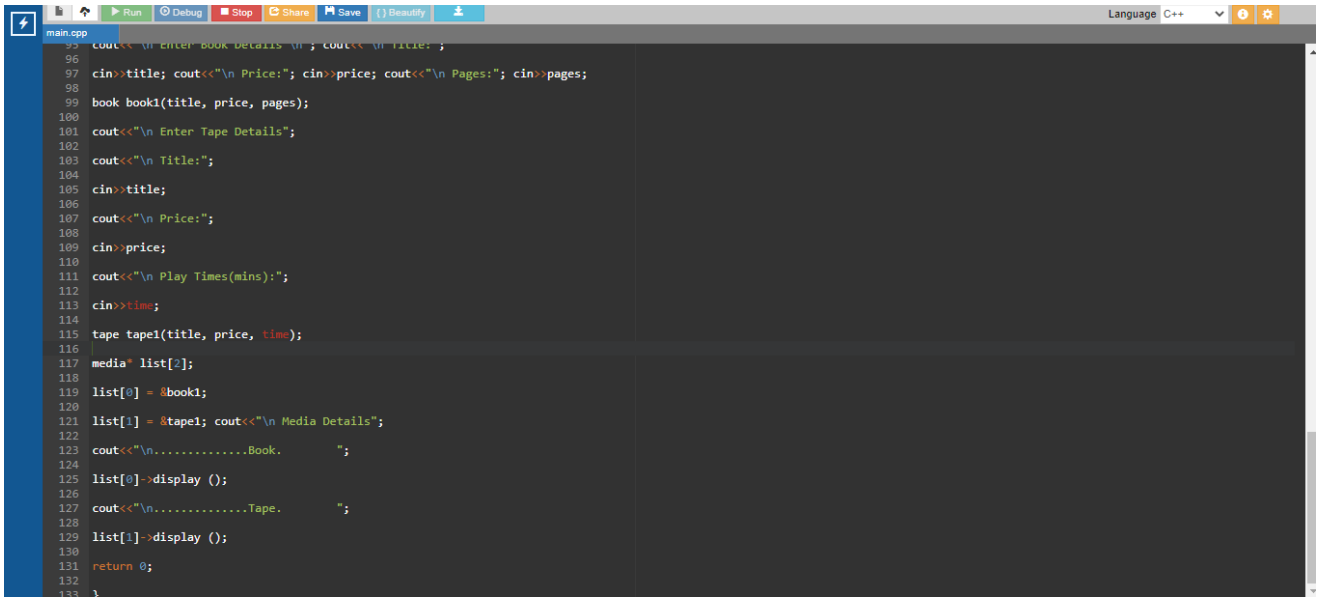
PROGRAM CODE-



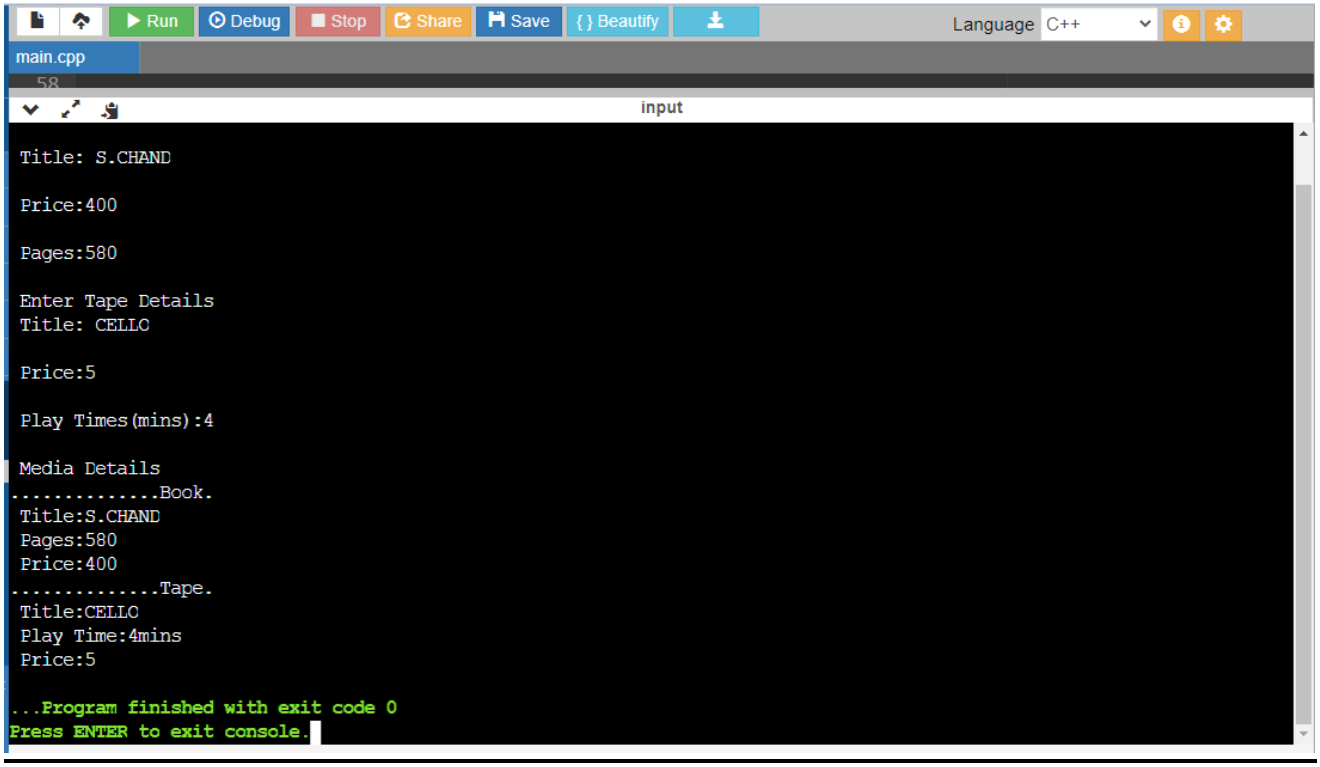
```
#include<iostream>
#include<string.h>
using namespace std;
class media
{
protected:
char title[30];
float price;
public:
media(char *s, float a)
{
strcpy(title, s); price = a;
}
virtual void display()()
};
class book : public media
{
int pages; public:
book(char *s, float a, int p) : media(s,a)
{
pages = p;
}
void display();
};
class tape : public media
{
float time; public:
tape(char * s, float a, float t):media(s,a)
{
time =t;
}
void display();
};
```



```
}
void display();
};
void book ::display()
{
cout<<"\n Title:"<<title;
cout<<"\n Pages:"<<pages; cout<<"\n Price:"<<price;
}
void tape ::display ()
{
cout<<"\n Title:"<<title;
cout<<"\n Play Time:"<<time<<"mins"; cout<<"\n Price:"<<price;
}
int main()
{
char * title = new char[30]; float price, time;
int pages;
cout<<"\n Enter Book Details \n"; cout<<"\n Title:";
cin>>title; cout<<"\n Price:"; cin>>price; cout<<"\n Pages:"; cin>>pages;
book book1(title, price, pages);
cout<<"\n Enter Tape Details";
cout<<"\n Title:";
cin>>title;
cout<<"\n Price:";
cin>>price;
cout<<"\n Play Times(mins):";
cin>>time;
tape tape1(title, price, time);
media* list[2];
list[0] = &book1;
list[1] = &tape1; cout<<"\n Media Details";
cout<<"\n.............Book.        ";
list[0]->display ();
```


Run
Debug
Stop
Share
Save
{ } Beautify
Language C++
main.cpp
cin>>title; cout<<"\n Price:"; cin>>price; cout<<"\n Pages:"; cin>>pages;
book book1(title, price, pages);
cout<<"\n Enter Tape Details";
cout<<"\n Title:";
cin>>title;
cout<<"\n Price:";
cin>>price;
cout<<"\n Play Times(mins):";
cin>>time;
tape tape1(title, price, time);
media* list[2];
list[0] = &book1;
list[1] = &tape1; cout<<"\n Media Details";
cout<<"\n.....Book. ";
list[0]->display ();
cout<<"\n.....Tape. ";
list[1]->display ();
return 0;

OUTPUT-


Run
Debug
Stop
Share
Save
{ } Beautify
Language C++
main.cpp
input
Title: S.CHAND
Price:400
Pages:580
Enter Tape Details
Title: CELLO
Price:5
Play Times(mins):4
Media Details
.....Book.
Title:S.CHAND
Pages:580
Price:400
.....Tape.
Title:CELLO
Play Time:4mins
Price:5
...Program finished with exit code 0
Press ENTER to exit console.

LEARNING OUTCOMES

- Identify situations where computational methods would be useful.
- Approach the programming tasks using techniques learnt and write pseudo-code.
- Choose the right data representation formats based on the requirements of the problem.
- Use the comparisons and limitations of the various programming constructs and choose the right one for the task.

EVALUATION COLUMN (To be filled by concerned faculty only)

Sr. No.	Parameters	Maximum Marks	Marks Obtained
1.	Worksheet Completion including writing learning objective/ Outcome	10	
2.	Post Lab Quiz Result	5	
3.	Student engagement in Simulation/ Performance/ Pre Lab Questions	5	
4.	Total Marks	20	