# EXPERIMENT NUMBER –Practical 7.1

**TOPIC OF 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.

**AIM OF THE EXPERIMENT**

Programs based on Run-time polymorphism.

**FLOWCHART/ ALGORITHM**

START

Step 1→ Creating a header file for input output stream and define the context.

Step 2→Creating class name followed by class Record and Declare data members such as name and age.

Step 3 → Within the class record defining data members inbound with private access specifier mode and member function inbound with public access specifier mode. Which can easily to get output out of the class.

Step 4 → Defining the member functions to get and show data in the output screen.

Step 5 → Defining the overloaded function to compare the ages of the two.

Step 6 → Creating the instance of the class and calling the required functions.

Step 8 → Printing the output in the console screen.

Step 9 →At the end output on the screen is who is greater this name and age will be printed on the screen.

stop

**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<<"==| smita shinde uid -20BCS4643 |=="<<endl;

cout<<name<<" : "<<age<<endl;

}

Records eldest(Records o)

{

return (o.age>age)? o: \*this;

}

};

int main()

{

Records ob[3]={Records("smita",19),Records("Dhruva",19),Records("Radhika",30)};

Records res;

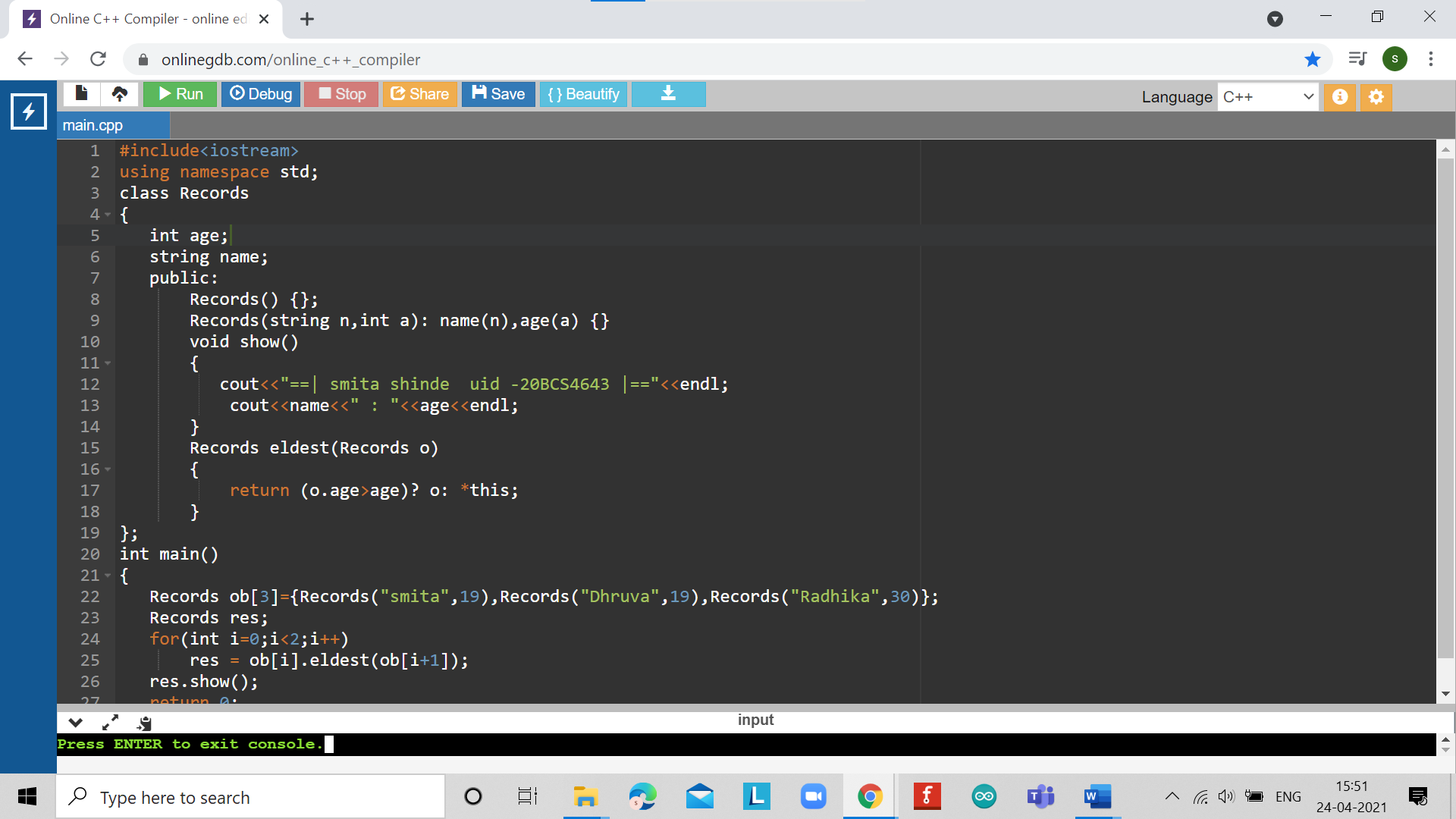
for(int i=0;i<2;i++)

res = ob[i].eldest(ob[i+1]);

res.show();

return 0;

}



**ERRORS ENCOUNTERED DURING PROGRAM’S EXECUTION**

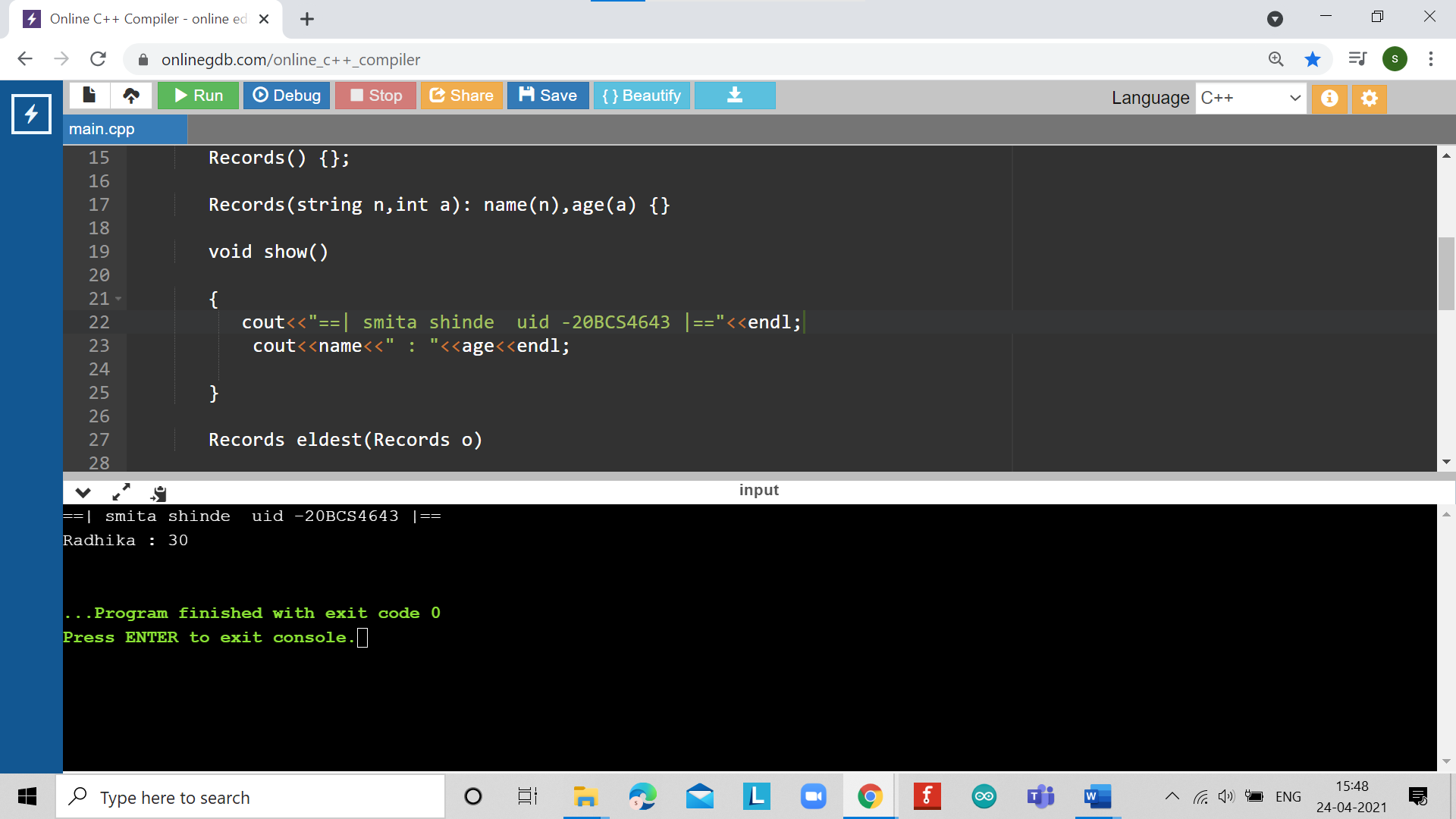
**(Kindly jot down the compile time errors encountered)**

No error while execution of the program**.**

**PROGRAMS’ EXPLANATION (in brief)**

1. We start our program with preprocessor (#) and header file (< iostream >) we have many types of header files but in this program, we used iostream. #include< iostream >, Int main () is a function which work as a container of statements. All the statements are enclosed within the pair of braces { }. “using namespace std” means we use the namespace named std. “std” is an abbreviation for standard. So that means we use all the things with in “std” namespace.
2. In this program we creating a class that will maintain the records of persons with name and age and finding eldest among them after creating a class record inside it we declaring data member such as name age then we declaring main function inside it declaring record and initializing record of three persons after that we checking conditions of eldest one then for loop will be in work, the number of looping equal to number of person’s data, we will pass argument to the function whose return type is Records, and then we will get the name who is oldest and his age. And displaying the eldest among them in console.
3. In short, the above programs take name and ages as input compares them and returns the greatest among them using operator overloading. It also uses this statement to return the output so as to show the run time polymorphism.

**OUTPUT**



# EXPERIMENT NUMBER –Practical 7.2

**TOPIC OF EXPERIMENT –**

WAP to access members using pointer to object members.

**AIM OF THE EXPERIMENT**

Programs based on Run-time polymorphism.

**FLOWCHART/ ALGORITHM**

START

Step 1→ Creating a header file for input output stream and define the context.

Step 2→Creating a class num along with a data member function and a constructor with a default value.

Step 3→Defining the member functions to get the number from the user and print it in the last in the required format.

Step 4→Creating the instance of the class along with a pointer of same class.

Step 5→Using “new” to create the object at run time.

Step 6→Calling the functions using the pointer.

Step 7→ Printing the output.

STOP

**PROGRAM CODE**

#include <iostream>

using namespace std;

class Number

{

private:

int num;

public:

//constructor

Number(){ num=0; };

//member function to get input

void inputNumber (void)

{

cout<<"==| smita shinde uid -20BCS4643 |=="<<endl;

cout<<"Enter an integer number: ";

cin>>num;

}

//member function to display number

void displayNumber()

{

cout<<"Num: "<<num<<endl;

}

};

//Main function

int main()

{

//declaring object to the class number

Number N;

//input and display number using norn object

N.inputNumber();

N.displayNumber();

//declaring pointer to the object

Number \*ptrN;

ptrN = new Number; //creating & assigning memory

//printing default value

cout<<"Default value... "<<endl;

//calling member function with pointer

ptrN->displayNumber();

//input values and print

ptrN->inputNumber();

ptrN->displayNumber();

return 0;

}

**ERRORS ENCOUNTERED DURING PROGRAM’S EXECUTION**

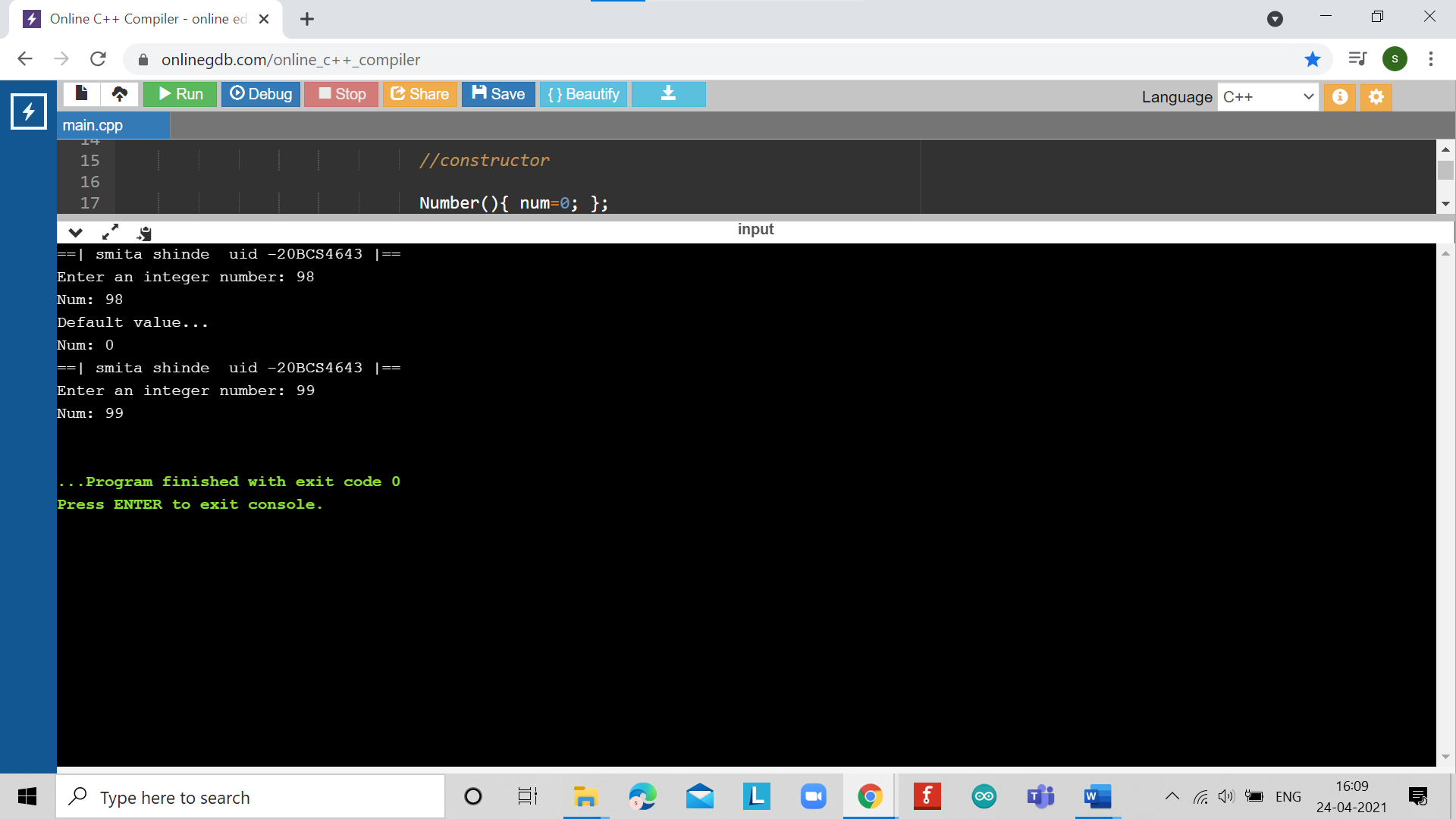
**(Kindly jot down the compile time errors encountered)**

No error occurred while execution of the program.

**PROGRAMS’ EXPLANATION (in brief)**

1. We start our program with preprocessor (#) and header file (< iostream >) we have many types of header files but in this program, we used iostream. #include< iostream >, Int main () is a function which work as a container of statements. All the statements are enclosed within the pair of braces { }. “using namespace std” means we use the namespace named std. “std” is an abbreviation for standard. So that means we use all the things with in “std” namespace.
2. In this program we accessing members using pointer to object member firstly we creating a class number as private access specifier then we we creating a constructor and initialize data member after that we declaring a member void number function to get input and declaring another member void display number to display output then we creating a main function for simple access the functions we will create a normal class object that will call the function in the class by ‘.’ dot operator and declaring a pointer object of class number type Assigning new number memory created at run time using ‘new’ to pointer variable then we Calling member functions and returning numbers after that displaying numbers in output screen**.**
3. In short,above programs takes an integer value from the user and the first prints it using normal object invocation then it prints the value of constructor. At last, it creates a new object for the pointer in the runtime and again calls the member function using that pointer**.**

**OUTPUT**



# EXPERIMENT NUMBER –Practical 7.3

**TOPIC OF 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.

**AIM OF THE EXPERIMENT**

Programs based on Run-time polymorphism.

**FLOWCHART/ ALGORITHM**

START

Step 1→ Creating a header file for input output stream and define the context.

Step 2→ Creating a class media with all the data members that are required.

Step 3→ Creating two classes book and tape separately and publicly inheriting media in both of them as virtual base class.

Step 4→ Defining the member functions outside the class to get input and to show the output database.

Step 5→ Creating separate pointers for both the class book and tape.

Step 6→ Using “new” keyword to create objects at run time for both.

Step 7→ Calling the member functions for both the classes.

STOP.

**PROGRAM CODE**

#include<iostream>

#include<string.h>

using namespace std;

class media

{

protected:

char title[50];

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 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<<"==| smita shinde uid -20BCS4643 |=="<<endl;

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 ();

cout<<"\n..............Tape. ";

list[1]->display ();

return

**ERRORS ENCOUNTERED DURING PROGRAM’S EXECUTION**

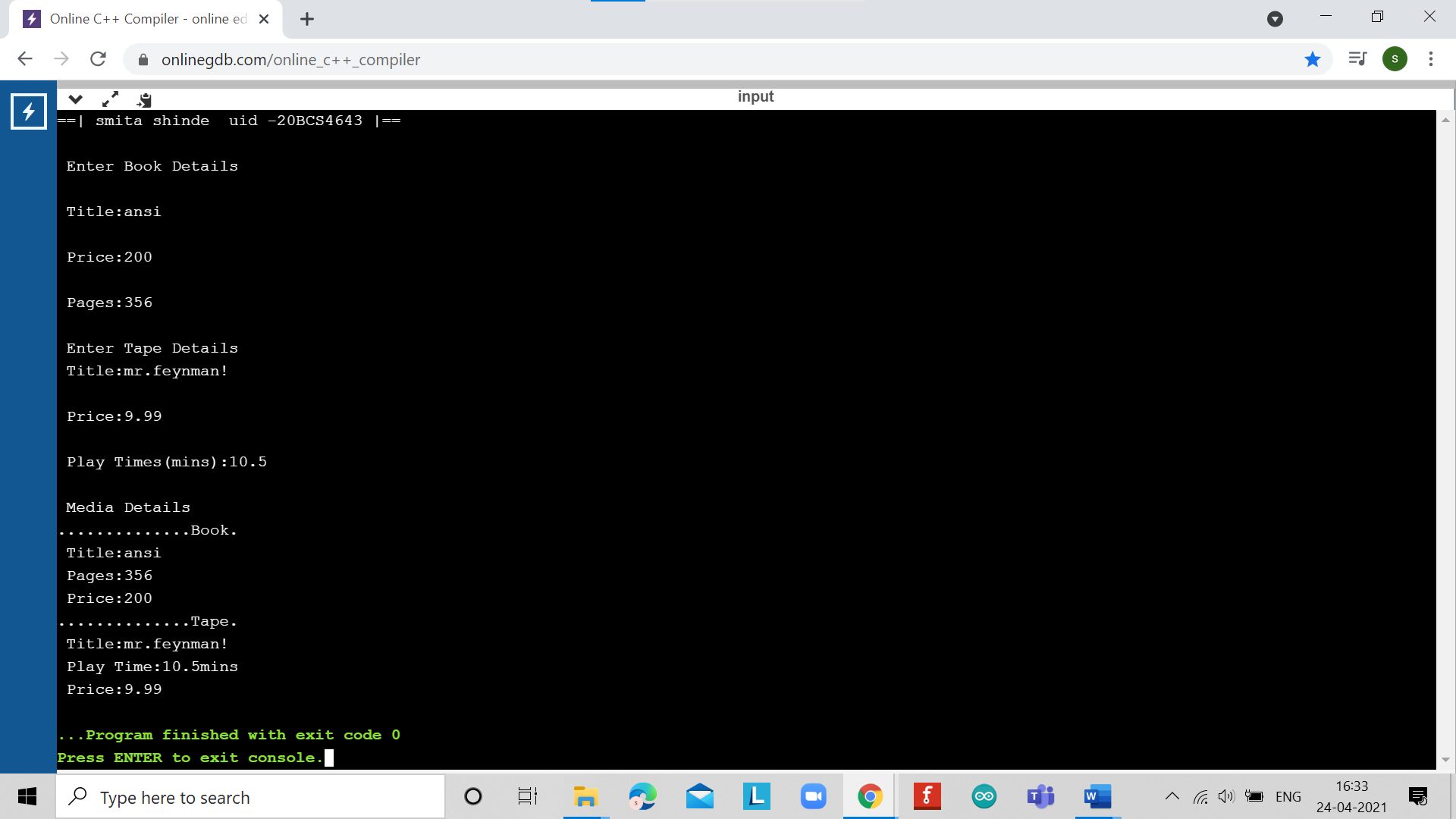
**(Kindly jot down the compile time errors encountered)**

No error occurred while execution of the program.

**PROGRAMS’ EXPLANATION (in brief)**

1. We start our program with preprocessor (#) and header file (< iostream >) we have many types of header files but in this program, we used iostream. #include< iostream >, Int main () is a function which work as a container of statements. All the statements are enclosed within the pair of braces { }. “using namespace std” means we use the namespace named std. “std” is an abbreviation for standard. So that means we use all the things with in “std” namespace.
2. In this program we designing a class which will represent the information regarding digital library such as books, tape: book & tape should be separate classes having the base class as media. Firstly we creating a base class media with the required variables in private such as title and price.Usinging the strcpy function of string we copies one string to another and Next two classes named book and tape will inherit the base class media with some certain required different variables such as in book pages will be added and in tape time will be added and initialization of variables will be done with constructor for each classes and the display will be done by friend or member function.In main function we will enter the name, price time and pages and according to the function argument we will pass these initialized variables to their respective functions after that we displaying output in console.
3. The above program creates a database of books and tapes using inheritance and “new” keyword. It stores price, title and pages for book and title, price and play time for the tapes. It creates the objects at runtime for the pointers hat are calling the member functions and at last it prints the output in desired format.

**OUTPUT**



**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 |  |