OOP Assignment 4

**Submitted By : Deepjyoti Deka – 190103014 - 4th Semester (CSE)**

1. Write a program in C++ to highlight the difference between overloaded assignment operator and copy constructor.

Ans:

Code:

#include <iostream>

using namespace std;

class circle

{

private:

int r;

float x, y;

public:

circle()

{

}

circle(int rr, float xx, float yy)

{

r = rr;

x = xx;

y = yy;

}

circle operator = (const circle &c)

{

cout << endl << "Assignment operator invoked";

r = c.r;

x = c.x;

y = c.y;

return circle(r, x, y);

}

circle(const circle &c)

{

cout << endl << "copy constructor invoked";

r = c.r;

x = c.x;

y = c.y;

}

void showdata()

{

cout << endl << "r = " << r;

cout << endl << "X=" << x;

cout << endl << "Y=" << y;

}

};

int main()

{

circle c1(15, 2.5, 2.5);

circle c2, c4;

c4 = c2 = c1;

circle c3 = c1;

c1.showdata();

c2.showdata();

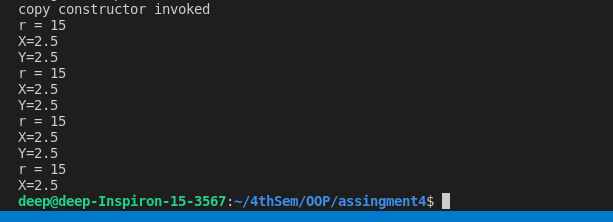
c3.showdata();

c4.showdata();

return 0;

}

Input/Output:



2.Write a Program illustrating how the constructors are implemented and the order in which they are called when the classes are inherited. Use three classes named alpha, beta, gamma such that alpha,beta are base class and gamma is derived class inheriting alpha & beta.

Ans:

Code:

#include <iostream>

using namespace std;

class alpha

{

int x;

public:

alpha(int i)

{

x = i;

cout << "alpha initialized\n";

}

void show\_x(void)

{

cout << "x=" << x << "\n";

}

};

class beta

{

float y;

public:

beta(float j)

{

y = j;

cout << "beta initialized\n";

}

void show\_y(void)

{

cout << "y= " << y << "\n";

}

};

class gamma : public beta, public alpha

{

int m, n;

public:

gamma(int a, float b, int c, int d) : alpha(a), beta(b)

{

m = c;

n = d;

cout << "gamma initialized\n";

}

void show\_mn(void)

{

cout << "m=" << m << "\n";

cout << "n=" << n << "\n";

}

};

int main()

{

gamma g(5, 10.75, 20, 30);

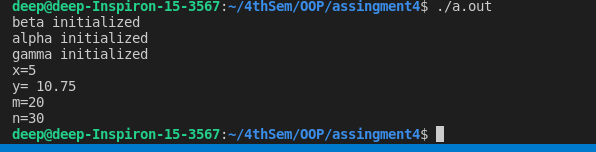
g.show\_x();

g.show\_y();

g.show\_mn();

}

Input/Output:



3.Write a Program to design a student class representing student roll no. and a test class (derived class of student) representing the scores of the student in various subjects and sports class representing the score in sports. The sports and test class should be inherited by a result class having the functionality to add the scores and display the final result for a student.

Ans:   
Code:

#include <iostream>

using namespace std;

class student

{

protected:

int roll;

public:

void get\_number(int a)

{

roll = a;

}

void put\_number(void)

{

cout << "Roll No:" << roll << "\n";

}

};

class test : public student

{

protected:

float part1, part2;

public:

void get\_marks(float x, float y)

{

part1 = x;

part2 = y;

}

void put\_marks(void)

{

cout << "Marks "

<< "\n"

<< "part1 =" << part1 << "\n"

<< "part2 =" << part2 << "\n";

}

};

class sports

{

protected:

float score;

public:

void get\_score(float s)

{

score = s;

}

void put\_score(void)

{

cout << "Sports wt:" << score << "\n\n";

}

};

class result : public test, public sports

{

float total;

public:

void display(void);

};

void result ::display(void)

{

total = part1 + part2 + score;

put\_number();

put\_marks();

put\_score();

cout << "Total Score is :" << total << "\n";

}

int main()

{

result student\_1;

student\_1.get\_number(124);

student\_1.get\_marks(27.5, 33.0);

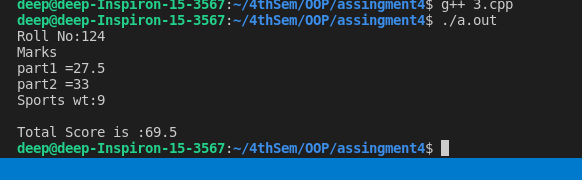
student\_1.get\_score(9.0);

student\_1.display();

return 0;

}

Input/Output:



4.Write a program to 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.

Ans:

Code:

#include <iostream>

#include <cstring>

using namespace std;

class human

{

char name[20];

float age;

public:

human(const char \*s, float a)

{

strcpy(name, s);

age = a;

}

human greater(human &x)

{

if (x.age >= age)

return x;

else

return \*this;

}

void display(void)

{

cout << "Name:" << name << "\n"

<< "Age: " << age << "\n";

}

};

int main()

{

human p1("deep", 7.50),p2("jyoti", 9.0),p3("ram", 10.5);

human p = p1.greater(p3);

cout << "The younger human is:\n";

p.display();

p = p1.greater(p2);

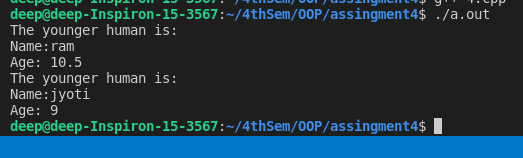
cout << "The younger human is:\n";

p.display();

return 0;

}

Input/Output:



5.Write a Program to illustrate the use of pointers to objects whch are related by inheritance.

Ans:

Code:

#include <iostream>

using namespace std;

class BC

{

public:

int b;

void show()

{

cout << "b=" << b << "\n";

}

};

class DC : public BC

{

public:

int d;

void show()

{

cout << "b=" << b << "\n"

<< "d=" << d << "\n";

}

};

int main()

{

BC \*bptr;

BC base;

bptr = &base;

bptr->b = 100;

cout << "bptr points to base object\n";

bptr->show();

DC derived;

bptr = &derived;

bptr->b = 200;

cout << "bptr now points to derived object\n";

bptr->show();

DC \*dptr;

dptr = &derived;

dptr->d = 300;

cout << "dptr is derived type pointer\n";

dptr->show();

cout << "Using ((DC \*)bptr)\n";

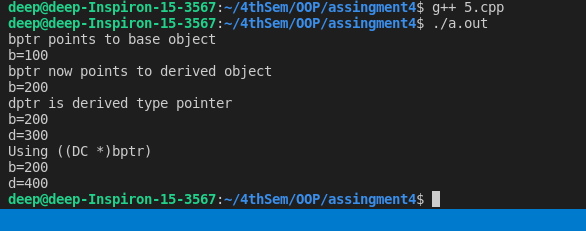
((DC \*)bptr)->d = 400;

((DC \*)bptr)->show();

return 0;

}

Input/Output:



6.Write a program illustrating the use of virtual functions in class.

Ans:

Code:

#include <iostream>

using namespace std;

class Base

{

public:

void display()

{

cout << "\n Display Base";

}

virtual void show()

{

cout << "\n Show Base:";

}

};

class Derived : public Base

{

public:

void display()

{

cout << "\n Display Derived";

}

void show()

{

cout << "\n Show Derived";

}

};

int main()

{

Base B;

Derived D;

Base \*bptr;

cout << "\n bptr points to Base\n";

bptr = &B;

bptr->display();

bptr->show();

cout << "\n\n bptr points to derived\n";

bptr = &D;

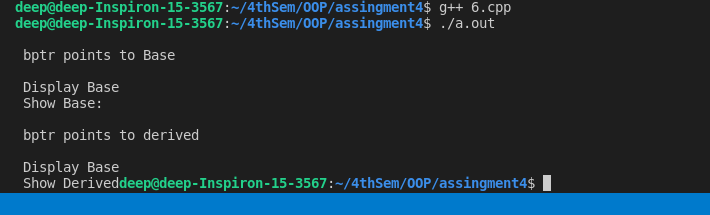
bptr->display();

bptr->show();

return 0;

}

Input/Output:



7.Write a program 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 use the runtime polymorphism.

Ans:  
Code:

#include <iostream>

#include <cstring>

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 << "\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();

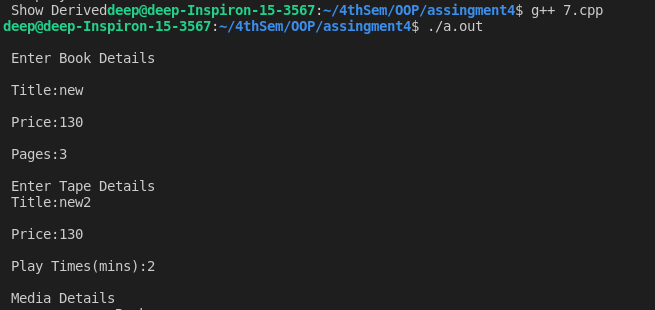
cout << "\n..............Tape.....";

list[1]->display();

return 0;

}

Input/Output:



8.Write a program to show conversion from string to int and vice-versa.

Ans:

Code:

#include<iostream>

#include <stdlib.h>

#include <string.h>

using namespace std;

class string

{

private:

char str[20];

public:

string ()

{

str[0] = '\0';

}

string (char \*s)

{

strcpy (str, s);

}

string (int a)

{

itoa (a, str, 10);

}

operator int ()

{

int i = 0, l, ss = 0, k = 1;

l = strlen (str) - 1;

while (l >= 0)

{

ss = ss + (str[l] - 48) \* k;

l--;

k \*= 10;

}

return (ss);

}

void displaydata ()

{

cout << str;

}

}

;

void

main ()

{

string s1 = 20;

cout << endl << "s1=";

s1.displaydata ();

s1 = 50;

cout << endl << "s1=";

s1.displaydata ();

string s2 ("20");

int i = int (s2);

cout << endl << "i=" << i;

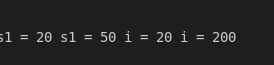
string s3 ("200");

i = s3;

cout << endl << "i=" << i;

}

Input/Output:



9.Write a program showing data conversion between objects of different classes.

Ans:

Code:

#include <bits/stdc++.h>

using namespace std;

class Time

{

int hrs, mins;

public:

Time(int, int);

operator int();

~Time()

{

cout << "Destructor is called." << endl;

}

};

Time::Time(int a, int b)

{

hrs = a;

mins = b;

}

Time::operator int()

{

cout << "Conversion of Class"

<< " Type to Primitive Type" << endl;

return (hrs \* 60 + mins);

}

void TypeConversion(int hour, int mins)

{

int duration;

Time t(hour, mins);

duration = t;

cout << "Total Minutes are " << duration << endl;

cout << "2nd method operator"

<< " overloading " << endl;

duration = t.operator int();

cout << "Total Minutes are " << duration << endl;

return;

}

int main()

{

int hour, mins;

hour = 2;

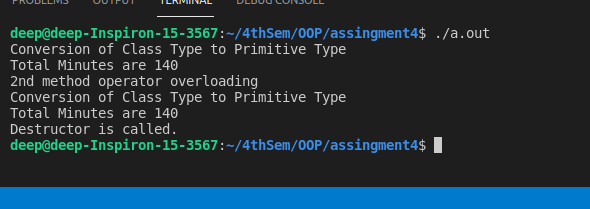
mins = 20;

TypeConversion(hour, mins);

return 0;

}

Input/Output:



10.Write a program showing data conversion between objects of different classes and conversion routine should reside in destination class.

Ans:

Code:

#include <bits/stdc++.h>

using namespace std;

class Time

{

int hrs, mins;

public:

Time(int, int);

operator int();

~Time()

{

cout << "Destructor is called." << endl;

}

};

Time::Time(int a, int b)

{

hrs = a;

mins = b;

}

Time::operator int()

{

cout << "Conversion of Class"

<< " Type to Primitive Type" << endl;

return (hrs \* 60 + mins);

}

void TypeConversion(int hour, int mins)

{

int duration;

Time t(hour, mins);

duration = t;

cout << "Total Minutes are " << duration << endl;

cout << "2nd method operator"

<< " overloading " << endl;

duration = t.operator int();

cout << "Total Minutes are " << duration << endl;

return;

}

int main()

{

int hour, mins;

hour = 3;

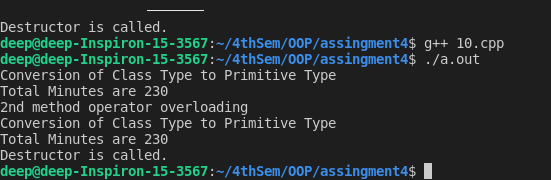
mins = 50;

TypeConversion(hour, mins);

return 0;

}

Input/Output:



11.Write a program to perform read/write binary I/O operation on a file

(i.e. write the object of a structure/class to file).

Ans:   
Code:

#include <fstream>

#include <iostream>

#include <string.h>

using namespace std;

int main(int argc, char \*\*argv)

{

int num;

char str[50];

cout << "Enter a String \n";

cin >> str;

int len = strlen(str);

cout << "Length of the string = " << len << endl;

fstream myFile("test.txt", ios::in | ios::out | ios::trunc);

myFile << str;

cout << "Enter the amount of characters to fetch: - \n";

cin >> num;

myFile.seekg(0, ios::beg);

char A[num];

myFile.read(A, num);

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

{

cout << A[i] << " ";

}

myFile.close();

return 0;

}

Input/Output:

