**Lab-2 Class & Objects**

1. WAP that reads two numbers. Calculate the smallest one and then print the smallest one defining functions outside the class.

2.WAP declaring a class and object to input assessment marks and practical marks defining functions setmark and publishmark. Define the function inside the class.

3. WAP input name, address, faculty and roll no of a student and display it. Use both member functions defining inside the class and outside the class.

#include<iostream>

using namespace std;

class student

{

private:

char name[20];

char address[20];

int roll;

char phone[20];

public:

void getdata()

{

cout<<"\nEnter student's name, address, roll and phone number";

cin>>name>>address>>roll>>phone;

}

void display()

{

cout<<"\n Student's name is="<<name;

cout<<"\n Student's address is="<<address;

cout<<"\n Student's roll is="<<roll;

cout<<"\n Student's phone is="<<phone;

}

};

int main()

{

student s;

s.getdata();

s.display();

return 0;

}

**OR**

#include<iostream>

#include<string.h>

using namespace std;

class student

{

private:

string name;

char address[20];

int roll;

char phone[20];

public:

void getdata()

{

cout<<"\nEnter student's name";

getline(cin,name);

cout<<"\naddress, roll and phone number";

cin>>address>>roll>>phone;

}

void display()

{

cout<<"\n Student's name is="<<name;

cout<<"\n Student's address is="<<address;

cout<<"\n Student's roll is="<<roll;

cout<<"\n Student's phone is="<<phone;

}

};

int main()

{

student s;

s.getdata();

s.display();

return 0;

}

4. Create a class called student with three data members (student name[20], roll\_no, Address) function called input-data() to take details of the students from the user, and a function called display\_data( )to display the details of the students. In main, create two objects of the class student and for each call both of the functions.

**/\* wap to read information of a student(name, address,roll,phone) and display them using function definition outside the class by passing arguments\*/**

/\* wap to read information of a student(name, address,roll,phone) and display them using function definition outside the class by passing arguments\*/

#include<iostream>

#include<string.h>

#include<stdint.h>

using namespace std;

class student

{

private:

char name[20];

char address[20];

int roll;

long long phone;

public:

void getdata(char x[],char y[],int z,long long w);

void display();

};

void student::getdata(char x[],char y[],int z, long long w)

{

strcpy(name,x);

strcpy(address,y);

roll = z;

phone=w;

}

void student::display()

{

cout<<"\n Student's name is="<<name;

cout<<"\n Student's address is="<<address;

cout<<"\n Student's roll is="<<roll;

cout<<"\n Student's phone is="<<phone;

}

int main()

{

student s;

char name[20];

char address[20];

cout<<"\n enter name"; cin>>name;

cout<<"\n Enter address"; cin>>address;

s.getdata(name,address, 5001, 9841392833);

s.display();

return 0;

}

5. Create a new class named city that will have two member variables Cityname(char[20]},and DistFromKTM(float). Add member functions to set and retrieve the CityName and DistFrom KTM separately. Add new member functions AddDistance that takes two arguments of class City and returns the sum of DistFrom KTM of two arguments. In main function,initialize three city objects. Set the first and second city to be Pokhara and Dhangadi. Display the sum of DistFromKTM of Pokhara and Dhangadi calling AddDistance function of third city object. **(use objects as function Arguments).**

#include<iostream>

#include<conio.h>

#include<string.h>

using namespace std;

class city

{ float distance;

char cityname[20];

float distfromktm;

public:

void getdata(char x[],float y)

{ strcpy(cityname,x);

distfromktm = y;

}

void display()

{

cout<<"City name is "<<cityname<<endl;

cout<<"Distance is "<<distfromktm<<endl;

}

void adddistance(city,city);

};

void city :: adddistance(city c1, city c2)

{ distance = c1.distfromktm + c2.distfromktm;

cout<<"total distance is="<<distance<<endl;

}

int main()

{

city s1,s2,s3;

s1.getdata("pokhara",250.50);

s2.getdata("dhangadi",350.56);

s1.display();

s2.display();

s3.adddistance(s1,s2);

return 0;

}

6. Write a program to read two integer numbers and display it. Use constructor to read the data.

#include <iostream>

using namespace std;

class param

{

int a, b;

public:

param (int, int);

void display ()

{

cout <<a<<" "<<b<<endl;

}

};

param:: param (int x, int y)

{

a = x;

b = y;

}

int main()

{

param p1 = param (10, 20); // expilit call

param p2(50,60);//implicit call

p1.display();

p2.display();

}

7. Write a program to make one object to read one integer data. Make another object and copy the data of first object to it. (Use the concept of copy constructor)

#include<iostream>

using namespace std;

class Copy

{

int id;

public:

Copy() { }// constructor

Copy(int x) //constructor

{

id = x;

}

Copy(Copy & y) //copy constructor

{

id = y.id;

}

void display()

{ cout<<id; }

};

int main()

{

Copy A (20); // object A is created & initialized

Copy B(A); // copy constructor called

Copy C = A; // copy constructor called again

Copy D; // D is created, not initialized

D = A; // copy constructor not called

cout<<"\n id of A: "; A.display();

cout<<"\n id of B: "; B.display();

cout<<"\n id of C: "; C.display();

cout<<"\n id of D: "; D.display();

}

8. Write a program to add two complex numbers using the concept of constructor overloading.

#include<iostream>

using namespace std;

class complex

{ float x,y;

public:

complex(){}

complex(float a){x=y=a; }

complex(float real, float imag)

{x= real; y= imag;}

void sum(complex,complex);

void show();

};

void complex::sum (complex c1, complex c2)

{ x = c1.x + c2.x;

y= c1.y + c2.y;

}

void complex::show()

{

cout<<x<<"+j"<<y<<"\n";

}

int main()

{

complex A(2.7,3.5);

complex B(1.6);

complex C;

C.sum(A,B);

cout<<"A="; A.show();

cout<<"B=";B.show();

cout<<"C=";C.show();

return 0;

}

9. Write a program to show the concept of a destructor.

#include <iostream>

using namespace std;

int count = 0;

class destruct

{

public:

destruct ()

{

count++;

cout <<"\n no. of object created is" << count;

}

~destruct ()

{

cout <<"\n no. of object destroyed" <<count;

count --;

}

};

int main()

{

destruct d1, d2, d3;

return 0;

}

10. Write a program to add two integer numbers and from two different objects and place it in third object and display the result. Use the concept of “returning objects from function”.

#include <iostream>

using namespace std;

class Example {

public:

int a;

Example add(Example Ea, Example Eb)

{

Example Ec;

Ec.a = Ea.a + Eb.a;

// *returning the object*

return Ec;

}

};

int main()

{

Example E1, E2, E3;

E1.a = 50;

E2.a = 100;

E3.a = 0;

cout << "Initial Values \n";

cout << "Value of object 1: " << E1.a

<< "\nobject 2: " << E2.a

<< "\nobject 3: " << E3.a

<< "\n";

// Passing object as an argument

// to function add()

E3 = E3.add(E1, E2);

cout << "\n\nNew values \n";

cout << "Value of object 1: " << E1.a

<< "\nobject 2: " << E2.a

<< "\nobject 3: " << E3.a

<< "\n";

return 0;

}

11. Write a program showing the concept of Static data member.

#include<iostream>

using namespace std;

class item{

static int X ;

public:

void get()

{

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

X++;

}

};

int item :: X;//definition of static data member. can also be initialized as

// int item :: count = 10;

int main()

{

item a,b,c;

a.get();

b.get();

c.get();

return 0;

}

12. Write a program to read name , age and salary of a employee using C++ structure.

#include <iostream>

using namespace std;

struct Person

{

char name[50];

int age;

float salary;

}; // *p1;*

int main()

{

Person p1; // *This is error in C. C declaration is struct Person p1;*

cout << "Enter Full name: ";

cin.get(p1.name, 50);

cout << "Enter age: ";

cin >> p1.age;

cout << "Enter salary: ";

cin >> p1.salary;

cout << "\nDisplaying Information." << endl;

cout << "Name: " << p1.name << endl;

cout <<"Age: " << p1.age << endl;

cout << "Salary: " << p1.salary;

return 0;

}