**Practice Questions:**

Q1)

Write a C++ program to find the area of circle using class circle which have following details:   
   
a. Accept radius from the user   
b. Calculate the area   
c. Display the result

Note: You should define all member functions inside the class.(refer example 3)

CODE;

#include <iostream>

using namespace std;

class circle

{

public:

int radius;

void input()

{

cin>>radius;

output();

}

private:

void output()

{

cout<<"area: "<<3.14\*radius\*radius;

}

};

int main()

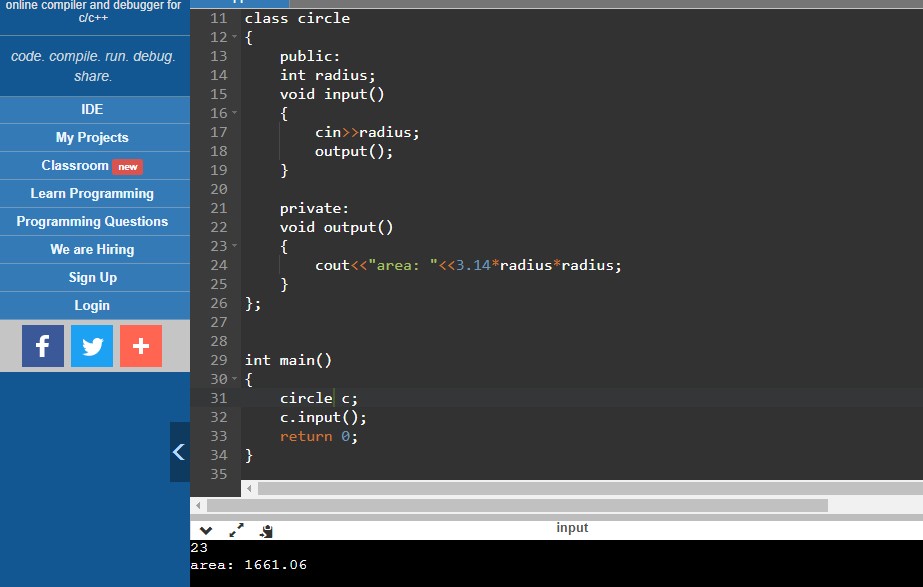
{

circle c;

c.input();

return 0;

}



Q2)

Create a class **art**with following members

**public:** a(int), b(int), c(float);

**private:** char d, double f;

**public:**get()- function to read  all values

**public:**display()-function to display  all values

write a main function  to read and display the  values.

CODE:

#include <iostream>

using namespace std;

class art

{

public:

int a,b;

float c;

private:

char d;

double f;

public:

void get()

{

cin>>a>>b>>c>>d>>f;

}

public:

void display()

{

get();

cout<<"a-"<<a<<" b-"<<b<<" c-"<<c<<" d-"<<d<<" f-"<<f;

}

};

int main()

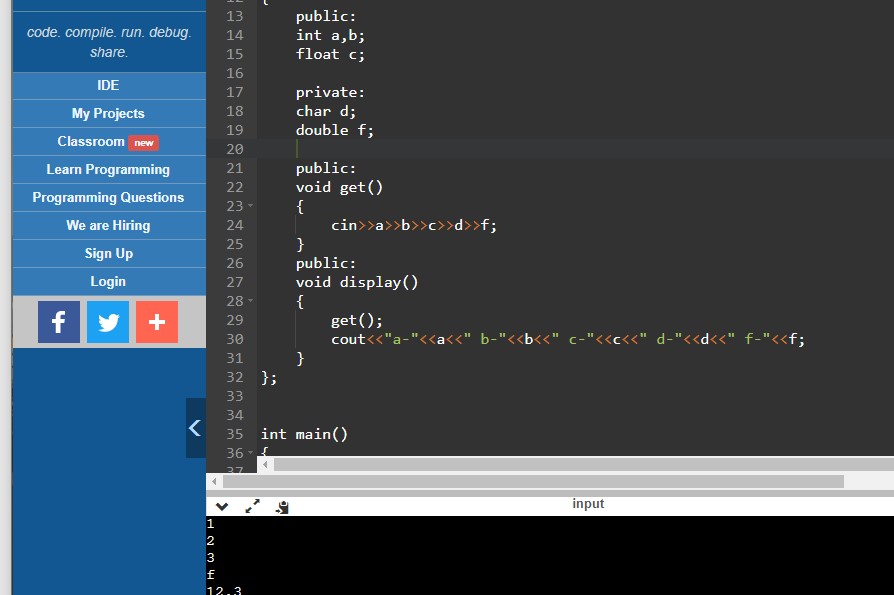
{

art c;

c.display();

return 0;

}



Q3)

Create a class **art**with following members

**public:** a(int), b(int), c(float);

**private:** char d, double f;

**private:**get()- function to read value

**public:**display()-function to display values

write a main function  to read and display the  values.

CODE:

#include <iostream>

using namespace std;

class art

{

public:

int a,b;

float c;

private:

char d;

double f;

void get()

{

cin>>a>>b>>c>>d>>f;

}

public:

void display()

{

get();

cout<<"a-"<<a<<" b-"<<b<<" c-"<<c<<" d-"<<d<<" f-"<<f;

}

};

int main()

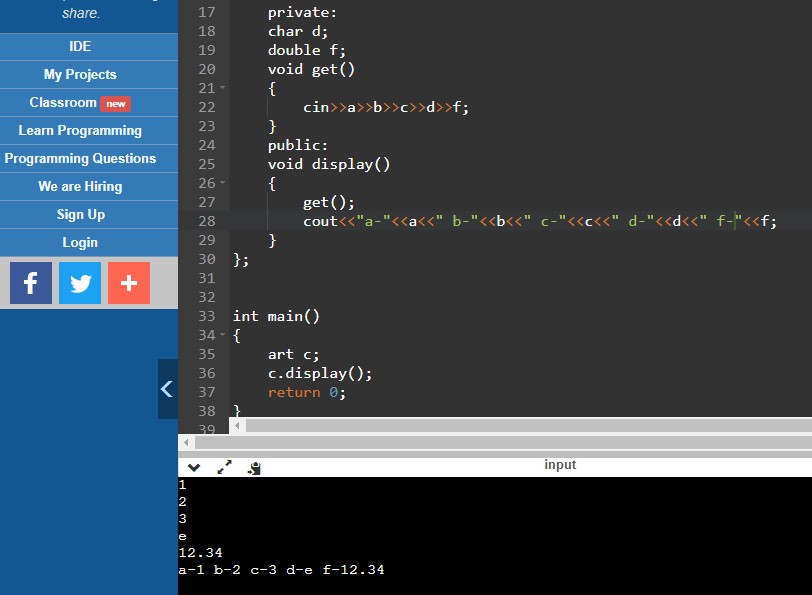
{

art c;

c.display();

return 0;

}



Q4)

Given the following class, called student, write the code for the following:

1)creates an object s1 of student

2)Initialize the member variables of s1 using studetails()  [ user Input ]

3) display the details  of s1 using display().

4) Complete main() function.

**Note:**

**There will be exactly one function call in main().**

**Finally your program should display the details of student s1.**

class student

{

**/\*\*  Find out the access modifiers of each members \*\*/**

char name[10];

int roll;

float mark;

studetails()--- give definition

display()---give definition

};

CODE:

#include <iostream>

using namespace std;

class student

{

char name[10];

int roll;

float mark;

public:

void studetails()

{

cin>>roll>>mark;

display();

}

private:

void display()

{

cout<<roll<<" "<<mark<<endl;

}

};

int main()

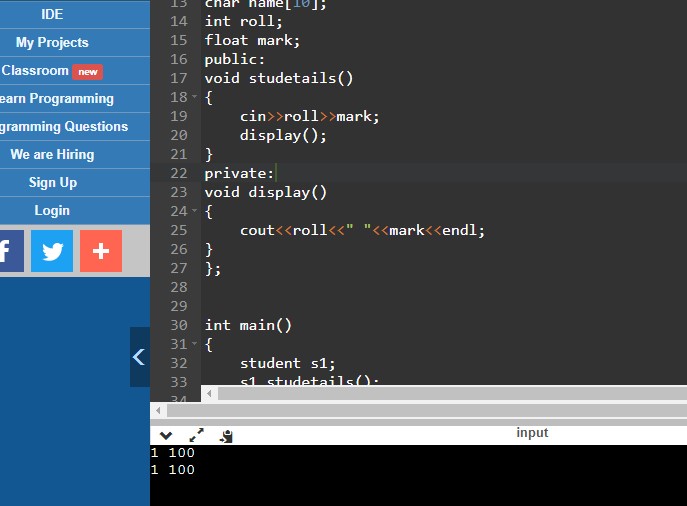
{

student s1;

s1.studetails();

return 0;

}



Q5)

Write a program to print the area of a rectangle by creating a class named 'Area' having two functions. First function named as 'setDim' takes the length and breadth of the rectangle from the user and the second function named as 'getArea' returns the area of the rectangle. Length and breadth of the rectangle are entered through keyboard.

**CODE:**

**#include <iostream>**

**using namespace std;**

**class area**

**{**

**public:**

**int length,breath;**

**void setDim()**

**{**

**cin>>length>>breath;**

**}**

**int setArea()**

**{**

**return length\*breath;**

**}**

**};**

**int main()**

**{**

**area a1;**

**int ar;**

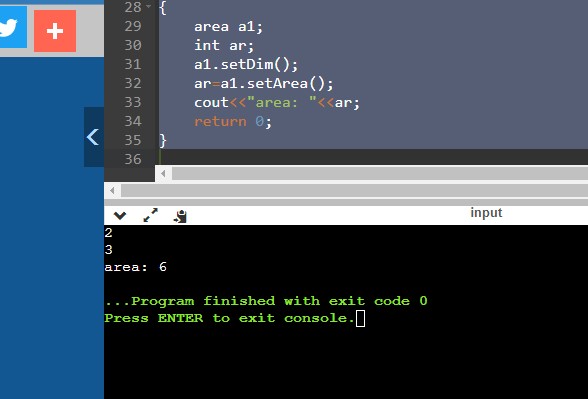
**a1.setDim();**

**ar=a1.setArea();**

**cout<<"area: "<<ar;**

**return 0;**

**}**

****

Q6)

Write a C++ program to define a class employee having members Emp-id, Emp-name, basic salary and functions accept() and display(). Calculate DA=25% of basic salary, HRA=800, I-tax=15% of basic salary. Display the payslip using appropriate output format.

Note: You should define all member functions outside the class (refer example 5)

CODE:

#include <iostream>

using namespace std;

class employee

{

public:

int empid;

char empname[10];

float salary;

void accept();

void display();

};

void employee::accept()

{

employee e2;

cin>>e2.empid;

cin>>e2.empname;

cin>>e2.salary;

}

void employee::display()

{

employee e3;

cout<<"employee details"<<endl<<"----------------"<<endl;

cout<<"empid : "<<e3.empid<<endl;

cout<<"empname : "<<e3.empname<<endl;

cout<<"salary : "<<e3.salary<<endl;

cout<<"DA : "<<(0.25\*e3.salary)<<endl;

cout<<"HRA : "<<800<<endl;

cout<<"--------------------------"<<endl;

cout<<"total salary : "<<(e3.salary-(0.25\*e3.salary+800+0.15\*e3.salary))<<endl;

cout<<"--------------------------";

}

int main()

{

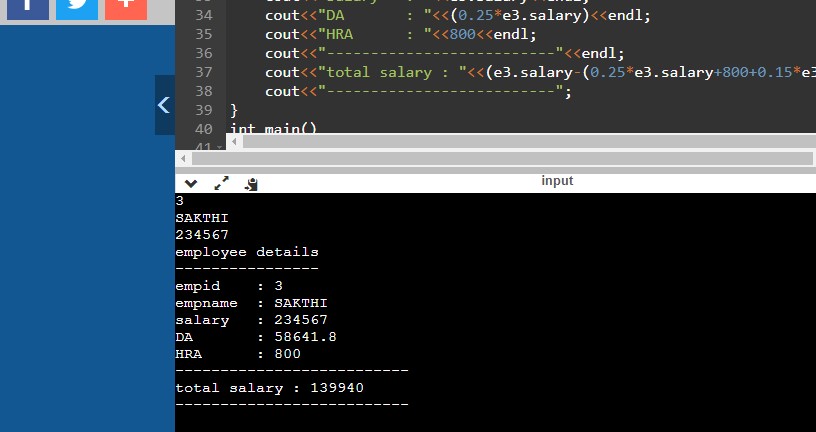
employee e1;

e1.accept();

e1.display();

return 0;

}



Q7)

Create a class point with  following members:

     Data members:

     public: x and  y coordinate

     Member functions:

     public:  getdata() to get the x, y coordinate values

     public: – checkquadrant () to find the quadrant in which the given coordinate lie.

     Write a c++  program  that reads x, y coordinate value from the user and display the     quadrant of the given point.

CODE:

#include <iostream>

using namespace std;

class point{

public:

int x,y,z;

void getdata()

{

cin>>x>>y;

}

point quadrant(point a)

{

point t;

if(a.x>0 && a.y>0)

{

t.z=1;

}

else if(a.x<0 && a.y>0)

{

t.z=2;

}

else if(a.x<0 && a.y<0)

{

t.z=3;

}

else if (a.x>0 && a.y<0)

{

t.z=4;

}

else

{

t.z=0;

}

return t;

}

};

int main()

{

point c;

c.getdata();

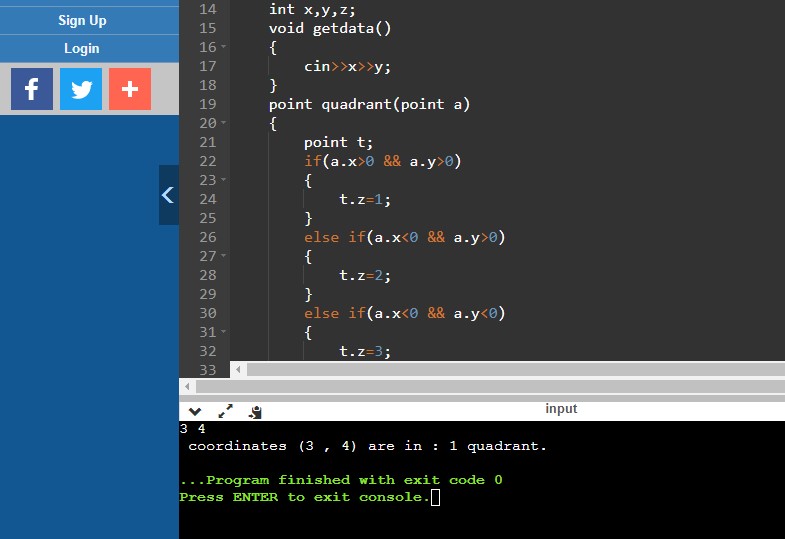
cout<<" coordinates ("<<c.x<<" , "<<c.y<<") are in : ";

point c1;

c1=c.quadrant(c);

cout<<c1.z<<" quadrant.";

}



Q8)

 Create a class **complex** with  following members:

     Data members:

     public: realpart and imagpart

     Member functions:

     Public-  getdata() to get real part and imaginary part

     public – add () to add  two complex numbers.

     public – sub () to subtract  two complex numbers.

     public – mul () to multiply  two complex numbers.

Write a c++ program  to add , subtract, multiply  two complex numbers .

 It is  a menu driven program in which a user will have to enter his/her choice to perform an operation.

Code:

#include <iostream>

using namespace std;

class Complex

{

public:

int real;

int imaginary;

int choice;

void getcomplex()

{

cin>>real;

cin>>imaginary;

}

void pick()

{

cout<<"enter 1-sum,2-subract,3-multiply"<<endl;

cin>>choice;

}

Complex addComplexNumber(Complex a, Complex b)

{

Complex res;

res.real = a.real + b.real;

res.imaginary = a.imaginary + b.imaginary;

cout << "Sum of complex number : "

<< res.real << " + i"

<< res.imaginary;

cout << endl

<< endl;

}

Complex subcomplexnumber(Complex a, Complex b)

{

Complex res;

res.real = a.real - b.real;

res.imaginary = a.imaginary - b.imaginary;

cout << "difference of complex number : "

<< res.real << " + i"

<< res.imaginary;

cout << endl

<< endl;

}

Complex multiplycno(Complex a, Complex b)

{

Complex res;

res.real = (a.real \* b.real) - (a.imaginary \* b.imaginary);

res.imaginary = (a.real \* b.imaginary) + (a.imaginary \* b.real);

cout << "product of complex number : "

<< res.real << " + i"

<< res.imaginary;

cout << endl

<< endl;

}

};

int main()

{

Complex C1, C2;

C1.getcomplex();

C2.getcomplex();

cout << "Complex number 1 : " << C1.real

<< " + i" << C1.imaginary << endl;

cout << "Complex number 2 : " << C2.real

<< " + i" << C2.imaginary << endl;

Complex C3;

error1:

C1.pick();

switch(C1.choice)

{

case 1:

C3= C1.addComplexNumber(C1, C2);

break;

case 2:

C3=C1.subcomplexnumber(C1,C2);

break;

case 3:

C3=C1.multiplycno(C1,C2);

break;

default: goto error1;

}

}

