**CODE:-**

#include <iostream>

using namespace std;

class num

{

public:

void my\_swap(int &m, int &n)

{

int temp;

temp = m;

m = n;

n = temp;

}

};

int main()

{

num n;

int a, b;

cout << "Enter the values of a and b." << endl;

cin >> a >> b;

cout << "Values of a and b are a: " << a << " b: " << b << endl;

n.my\_swap(a, b);

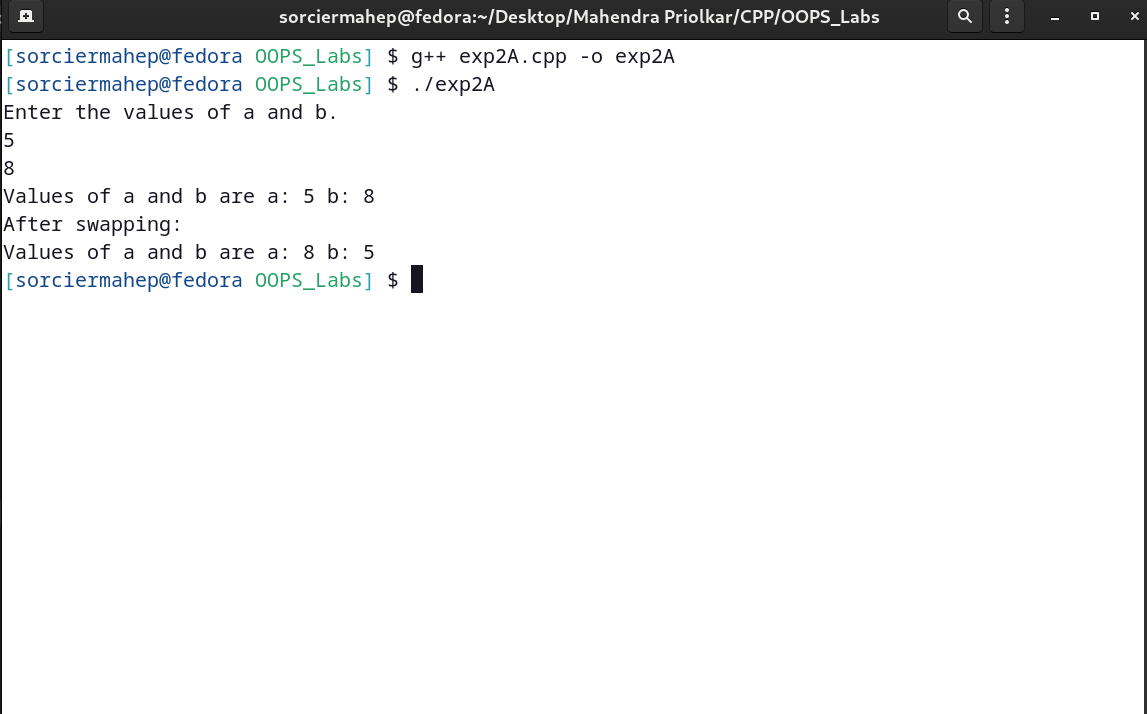
cout << "After swapping:" << endl;

cout << "Values of a and b are a: " << a << " b: " << b << endl;

return 0;

}

**OUTPUT:-**

****

**CODE:-**

#include <iostream>

#define PI 3.14159

using namespace std;

class calc

{

float r;

float s;

float circar;

public:

inline float area(int);

void get()

{

cout << "Enter radius of circle and side of square." << endl;

cin >> r >> s;

}

void calcar()

{

circar = area(r);

}

void display()

{

cout << "Area of circle is " << circar << "." << endl;

}

friend float sqarcalc(float);

};

float calc::area(int r)

{

float a = PI \* r \* r;

return a;

}

float sqarcalc(int s)

{

float a = s \* s;

return a;

}

int main()

{

calc c;

float sqarar;

c.get();

c.calcar();

c.display();

sqarar = sqarcalc(5);

cout << "Area of square is " << sqarar << "." << endl;

return 0;

}

**OUTPUT:-**



**CODE:-**

#include <iostream>

#define PI 3.14159

using namespace std;

class calc

{

float ca;

public:

void area(int s)

{

cout << "This figure is square." << endl;

ca = s \* s;

display();

}

void area(int l, int b)

{

cout << "This figure is rectangle." << endl;

ca = l \* b;

display();

}

void area(float r)

{

cout << "This figure is circle." << endl;

ca = PI \* r \* r;

display();

}

void display()

{

cout << "The area is : " << ca << endl;

}

};

int main()

{

calc c1, c2, c3;

int a = 5, b = 5, c = 10;

float d = 5.5;

cout << "Here, square and rectangle have integer dimensions and circle has float dimensions." << endl;

cout << "a=" << a << endl;

c1.area(a);

cout << "b=" << b << " "

<< "c=" << c << endl;

c2.area(b, c);

cout << "d=" << d << endl;

c3.area(d);

return 0;

}

**OUTPUT:-**

