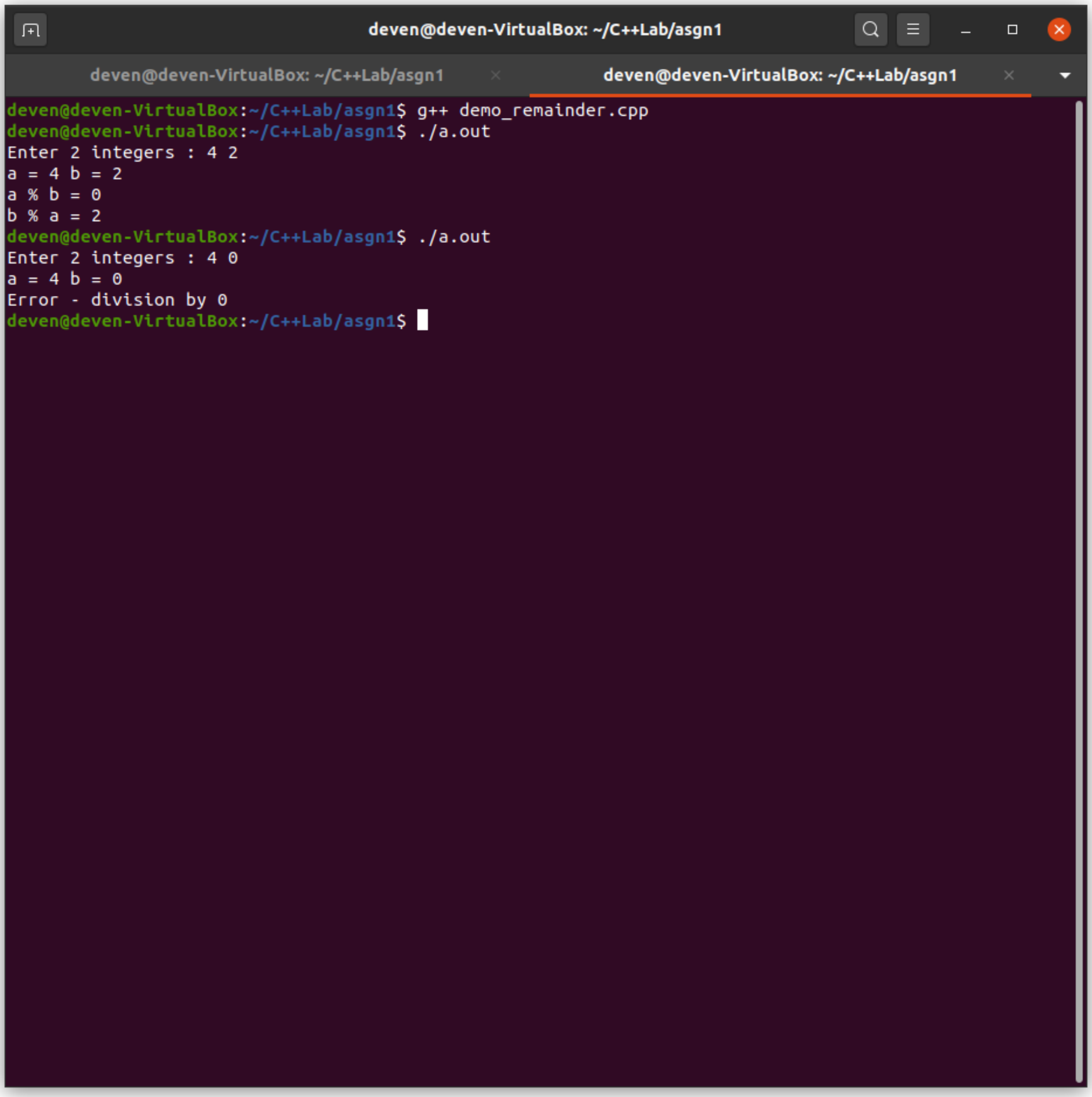
1. to demonstrate remainder operator

```
#include <iostream>
using namespace std;

int main()
{
        int a, b;
        cout << "Enter 2 integers : ";
        cin >> a >> b;
        cout << "a = " << a << " b = " << b <<endl;
        if(b == 0)
        {
                cout << "Error - division by 0 " << endl;
                exit(0);
        }
        cout << "a % b = " << a % b << endl;
        cout << "b % a = " << b % a << endl;
        return 0;
}
```

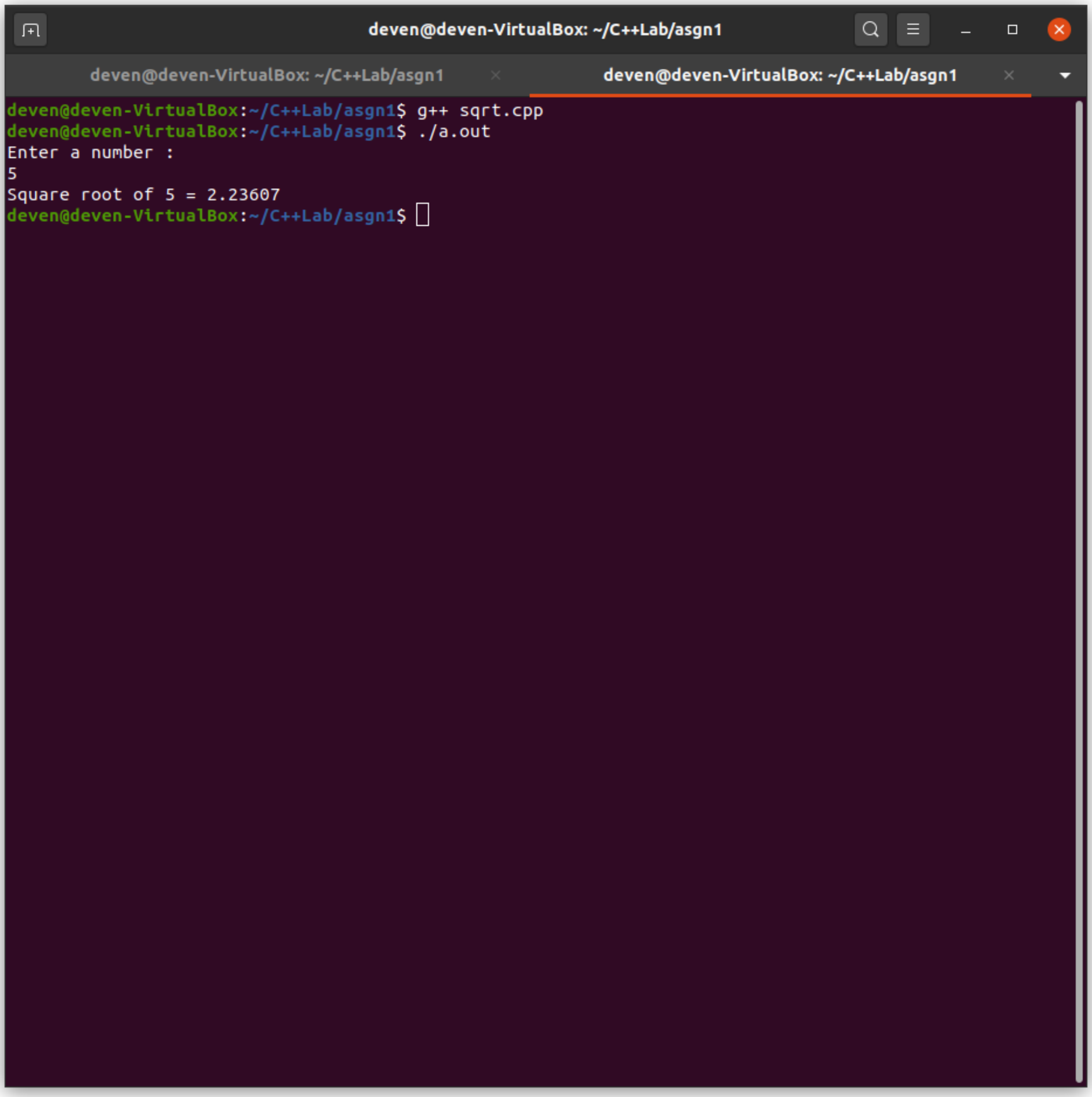


deven@deven-VirtualBox: ~/C++Lab/asgn1
deven@deven-VirtualBox: ~/C++Lab/asgn1
deven@deven-VirtualBox: ~/C++Lab/asgn1

```
deven@deven-VirtualBox:~/C++Lab/asgn1$ g++ demo_remainder.cpp
deven@deven-VirtualBox:~/C++Lab/asgn1$ ./a.out
Enter 2 integers : 4 2
a = 4 b = 2
a % b = 0
b % a = 2
deven@deven-VirtualBox:~/C++Lab/asgn1$ ./a.out
Enter 2 integers : 4 0
a = 4 b = 0
Error - division by 0
deven@deven-VirtualBox:~/C++Lab/asgn1$
```

2. to find sqrt of a number

```
#include <iostream>

#include <cmath>

using namespace std;


int main()

{

        double a;

        cout << "Enter a number : " << endl;
        cin >> a;

        cout << "Square root of " << a << " = " << sqrt(a) << endl;

        return 0;

}
```



deven@deven-VirtualBox: ~/C++Lab/asgn1
deven@deven-VirtualBox: ~/C++Lab/asgn1
deven@deven-VirtualBox: ~/C++Lab/asgn1

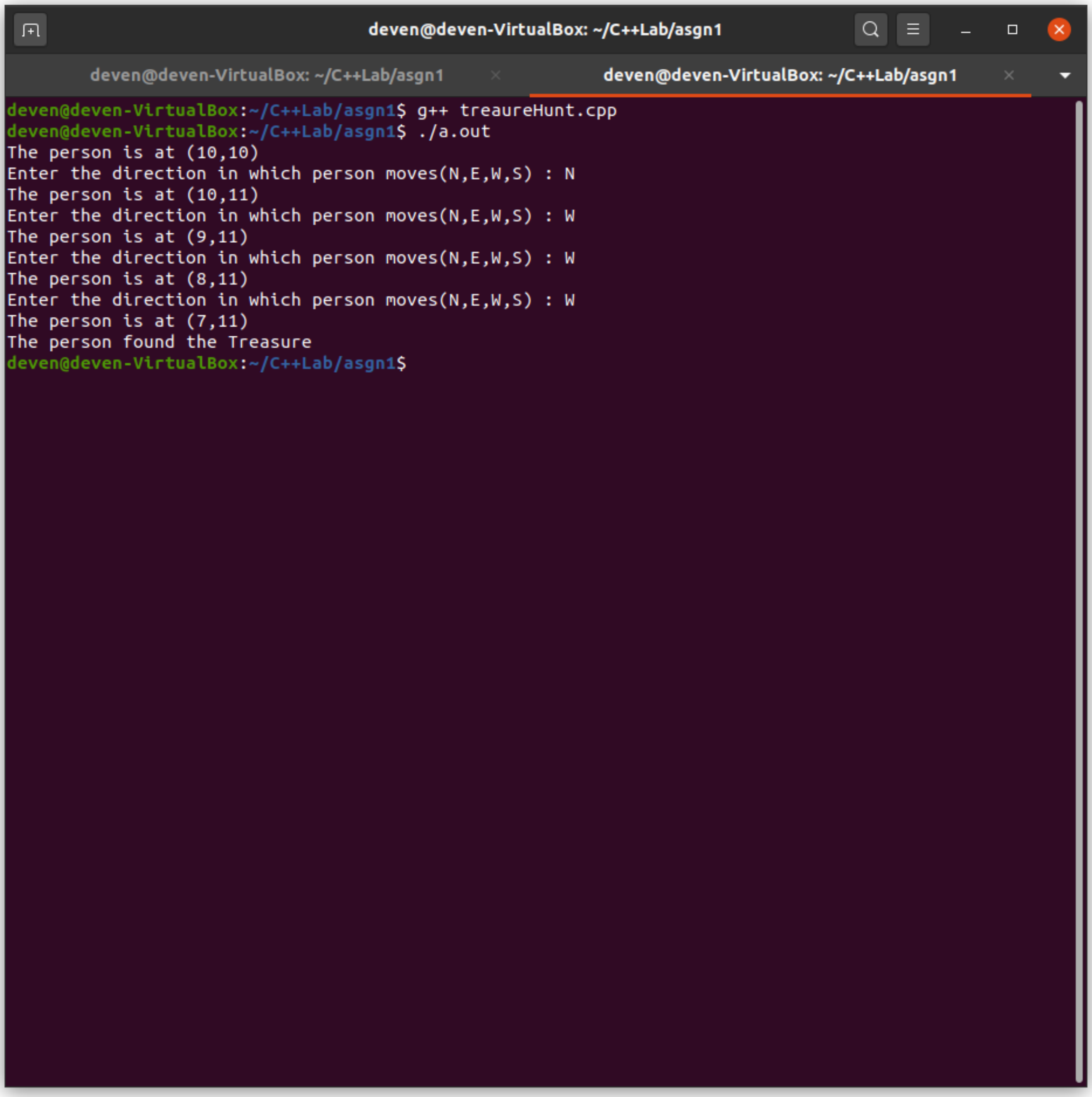
```
deven@deven-VirtualBox:~/C++Lab/asgn1$ g++ sqrt.cpp
deven@deven-VirtualBox:~/C++Lab/asgn1$ ./a.out
Enter a number :
5
Square root of 5 = 2.23607
deven@deven-VirtualBox:~/C++Lab/asgn1$ 
```

3. A person is at location (10,10), moves stepwise to directions based on (N, W, E, S). If the person reaches location (7,11) treasure is found, at location (5,11) a maneater is found . Use a switch case.

```
#include <iostream>
```

```
using namespace std;

int main()
{
	int x=10,y=10;
	char direction;
	cout<<"The person is at (10,10)"<<endl;
	while(true)
	{
		cout<<"Enter the direction in which person moves(N,E,W,S) : ";
		cin>>direction;
		switch(direction)
		{
			case 'N':
				y++;
				break;
			case 'E':
				x++;
				break;
			case 'W':
				x--;
				break;
			case 'S':
				y--;
				break;
			default:
				cout<<"Wrong input"<<endl;
		}
		cout<<"The person is at ("<<x<<","<<y<<")"<<endl;
		if(x==7 && y==11)
		{
```

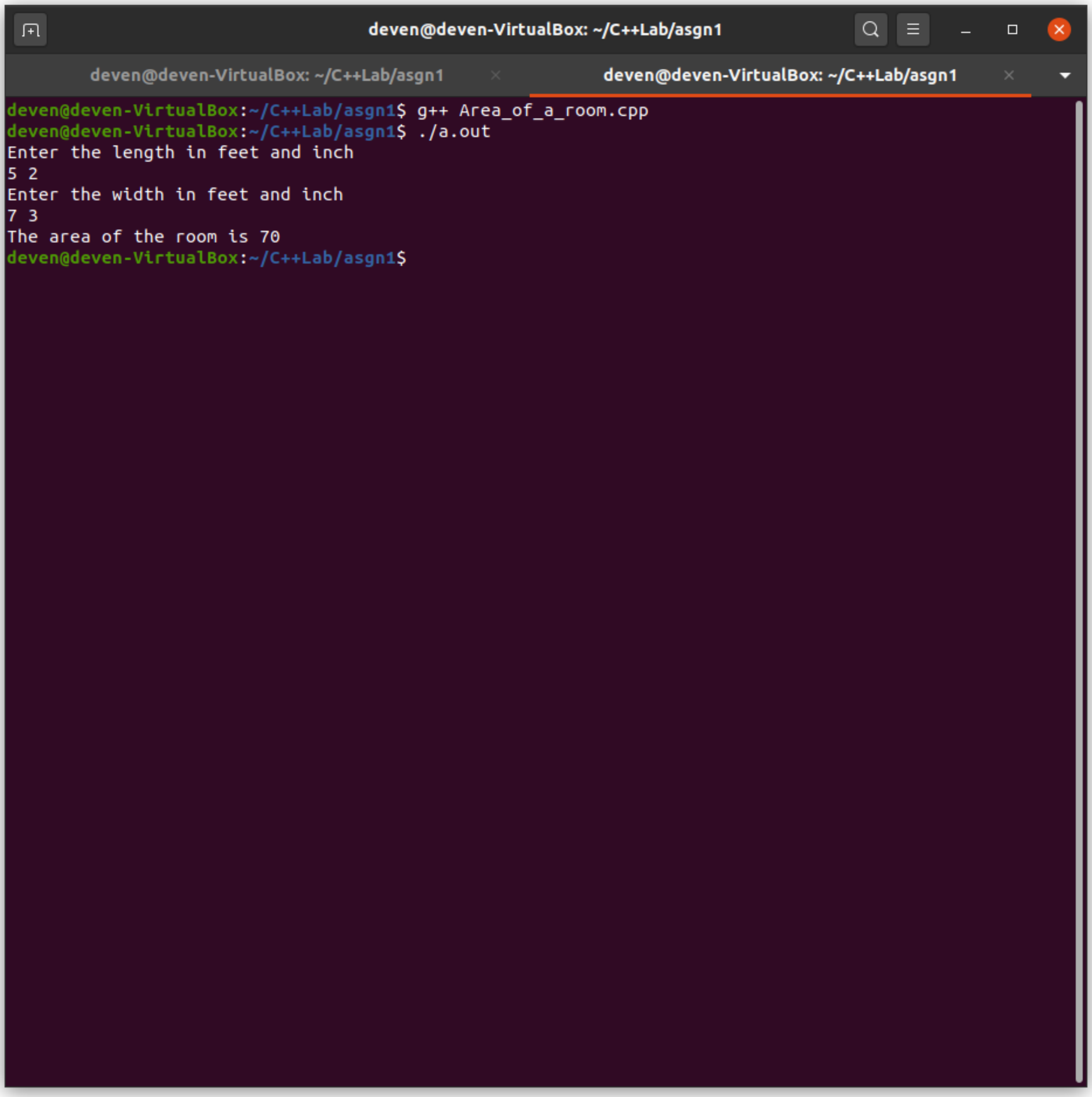
```
            cout<<"The person found the Treasure"<<endl;
            break;
        }
        if(x==5 && y==11)
        {
            cout<<"The person was eaten by the Maneater"<<endl;
            break;
        }
    }
    return 0;
}
```


deven@deven-VirtualBox: ~/C++Lab/asgn1
deven@deven-VirtualBox: ~/C++Lab/asgn1
deven@deven-VirtualBox: ~/C++Lab/asgn1

```
deven@deven-VirtualBox:~/C++Lab/asgn1$ g++ treaureHunt.cpp
deven@deven-VirtualBox:~/C++Lab/asgn1$ ./a.out
The person is at (10,10)
Enter the direction in which person moves(N,E,W,S) : N
The person is at (10,11)
Enter the direction in which person moves(N,E,W,S) : W
The person is at (9,11)
Enter the direction in which person moves(N,E,W,S) : W
The person is at (8,11)
Enter the direction in which person moves(N,E,W,S) : W
The person is at (7,11)
The person found the Treasure
deven@deven-VirtualBox:~/C++Lab/asgn1$
```

4. Consider two structures : Distance (feet and inches) and Room(length and width). For a dining room instance, calculate the area in square feet.

```
#include <iostream>

using namespace std;


struct Distance

{

        int feet;

        float inch;
};

struct Room
{
	Distance length;
	Distance width;
};

int main()
{
	Room dinning;
	float total_length,total_width;

	cout<<"Enter the length in feet and inch"<<endl;
	cin>>dinning.length.feet>>dinning.length.inch;
	cout<<"Enter the width in feet and inch"<<endl;
	cin>>dinning.width.feet>>dinning.width.inch;

	total_length=dinning.length.feet+dinning.length.inch;
	total_width=dinning.width.feet+dinning.width.inch;

	cout<<"The area of the room is "<<total_length*total_width<<endl;

	return 0;
}
```


deven@deven-VirtualBox: ~/C++Lab/asgn1
deven@deven-VirtualBox: ~/C++Lab/asgn1
deven@deven-VirtualBox: ~/C++Lab/asgn1

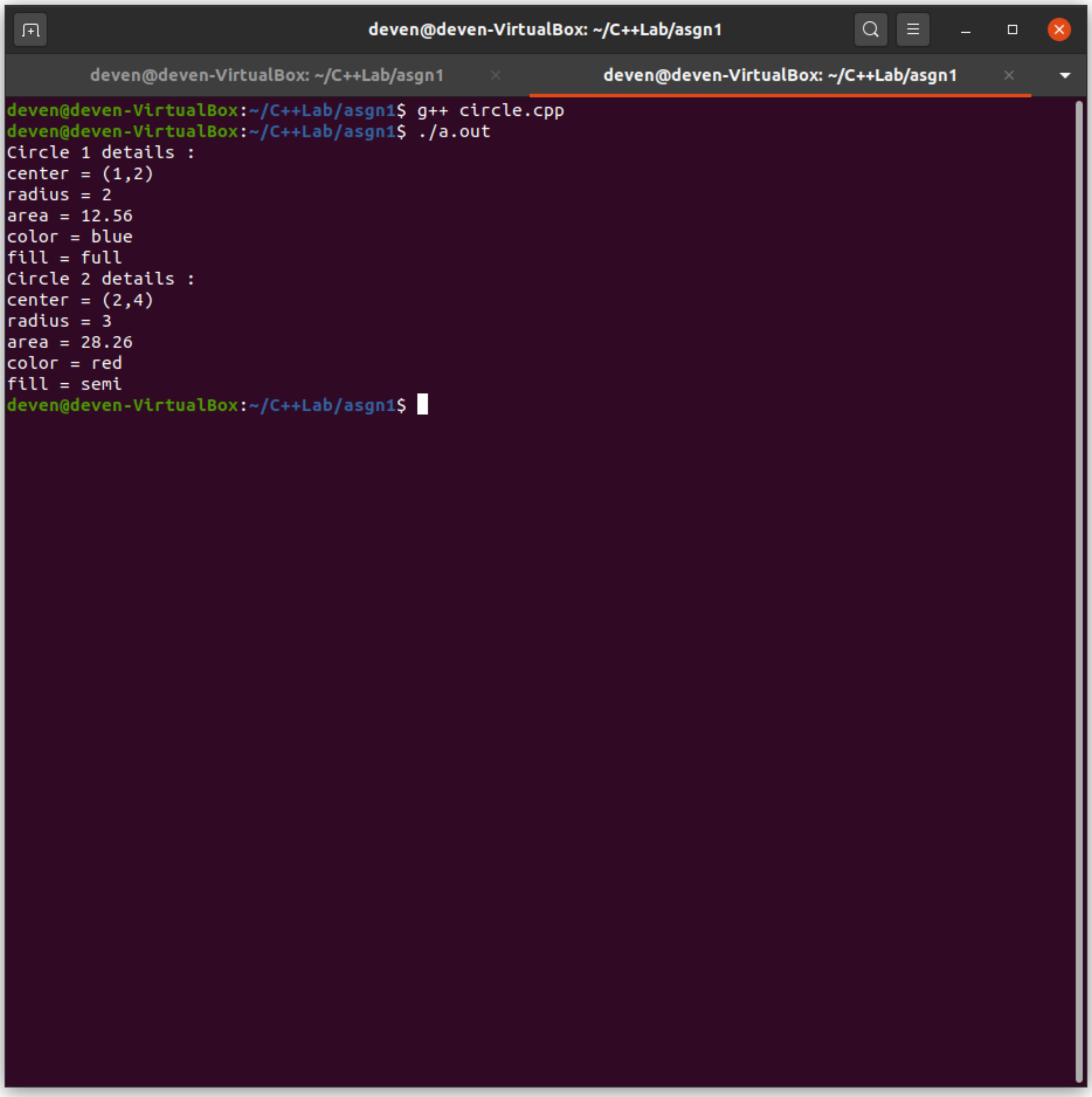
```
deven@deven-VirtualBox:~/C++Lab/asgn1$ g++ Area_of_a_room.cpp
deven@deven-VirtualBox:~/C++Lab/asgn1$ ./a.out
Enter the length in feet and inch
5 2
Enter the width in feet and inch
7 3
The area of the room is 70
deven@deven-VirtualBox:~/C++Lab/asgn1$
```

5. Create a structure circle, with suitable attributes and functions, assume (functions name only) different colors to draw the circle, and fill the circles [semi, full, void].

```
#include <iostream>

#include <string>

using namespace std;


struct point

{

        int x;
```

```
    int y;
};

struct circle
{
    point center;
    float radius;
    float area;
    string color;
    string fill;

    circle(point p,float r,string clr,string fi)//the variable names of classes(like string) used in the structure can't be used here as parameters, eg string color can't be used as a parameter, so changed to string clr
    {
        center=p;
        radius=r;
        area=3.14*r*r;
        color=clr;
        fill=fi;
    }
};

int main()
{
    circle c1({1,2},2,"blue","full");//if we were doing c1={{1,2},2,"blue","full"},i.e normal struct initialization, it wouldn't work coz, in that we have to pass all the arguments, i.e area value also we have to pass
    circle c2({2,4},3,"red","semi");
    cout<<"Circle 1 details : "<<endl;
    cout<<"center = ("<<c1.center.x<<","<<c1.center.y<<")"<<endl;
    cout<<"radius = "<<c1.radius<<endl;
```

```
        cout<<"area = "<<c1.area<<endl;
        cout<<"color = "<<c1.color<<endl;
        cout<<"fill = "<<c1.fill<<endl;
        cout<<"Circle 2 details : "<<endl;
        cout<<"center = ("<<c2.center.x<<","<<c2.center.y<<")"<<endl;
        cout<<"radius = "<<c2.radius<<endl;
        cout<<"area = "<<c2.area<<endl;
        cout<<"color = "<<c2.color<<endl;
        cout<<"fill = "<<c2.fill<<endl;
        return 0;
}
```



deven@deven-VirtualBox: ~/C++Lab/asgn1
deven@deven-VirtualBox: ~/C++Lab/asgn1
deven@deven-VirtualBox: ~/C++Lab/asgn1

```
deven@deven-VirtualBox:~/C++Lab/asgn1$ g++ circle.cpp
deven@deven-VirtualBox:~/C++Lab/asgn1$ ./a.out
Circle 1 details :
center = (1,2)
radius = 2
area = 12.56
color = blue
fill = full
Circle 2 details :
center = (2,4)
radius = 3
area = 28.26
color = red
fill = semi
deven@deven-VirtualBox:~/C++Lab/asgn1$ 
```