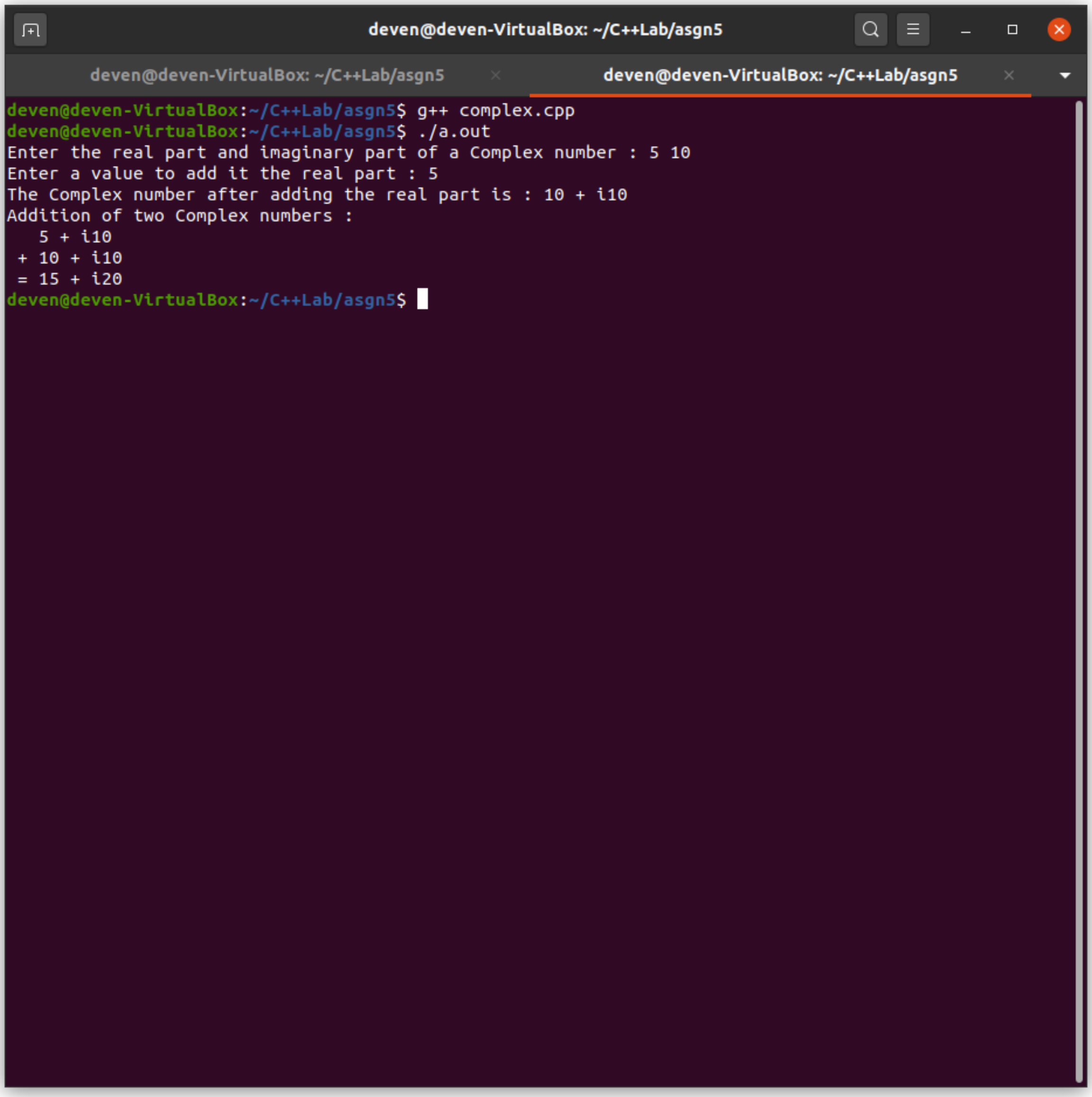
1.Write a C++ program to create a class called COMPLEX and implement the following overloading functions ADD that return a COMPLEX number.

i. ADD (a, s2) – where a is an integer (real part) and s2 is a complex number.

ii. ADD (s1, s2) – where s1 and s2 are complex numbers.

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

class Complex
{
    private:
        int x,y;
    public:
        Complex(int x_=0,int y_=0): x(x_), y(y_){}
        void get();
        Complex add(int a, Complex s2);
        Complex add(Complex s1, Complex s2);
        void print();
};

void Complex::get()
{
    cout<<"Enter the real part and imaginary part of a Complex number : ";
    cin>>x>>y;
}

Complex Complex::add(int a, Complex s2)
{
        Complex c(s2.x + a, s2.y);

        return c;

}


Complex Complex::add(Complex s1, Complex s2)

{

        Complex c(s1.x + s2.x, s1.y + s2.y);

        return c;

}


void Complex::print()

{

        cout<<x<<" + i"<<y<<endl;

}


int main()

{

        Complex c1;

        c1.get();

        int a;

        cout<<"Enter a value to add it the real part : ";

        cin>>a;

        Complex c2 = c1.add(a, c1);

        cout<<"The Complex number after adding the real part is : ";

        c2.print();

        Complex c3 = c1.add(c1, c2);
        cout<<"Addition of two Complex numbers : "<<endl;
        cout<<"   ";
        c1.print();
        cout<<" + ";
        c2.print();
        cout<<" = ";
        c3.print();
        return 0;
}
```



deven@deven-VirtualBox: ~/C++Lab/asgn5
deven@deven-VirtualBox: ~/C++Lab/asgn5
deven@deven-VirtualBox: ~/C++Lab/asgn5

```
deven@deven-VirtualBox:~/C++Lab/asgn5$ g++ complex.cpp
deven@deven-VirtualBox:~/C++Lab/asgn5$ ./a.out
Enter the real part and imaginary part of a Complex number : 5 10
Enter a value to add it the real part : 5
The Complex number after adding the real part is : 10 + i10
Addition of two Complex numbers :
   5 + i10
 + 10 + i10
 = 15 + i20
deven@deven-VirtualBox:~/C++Lab/asgn5$ 
```

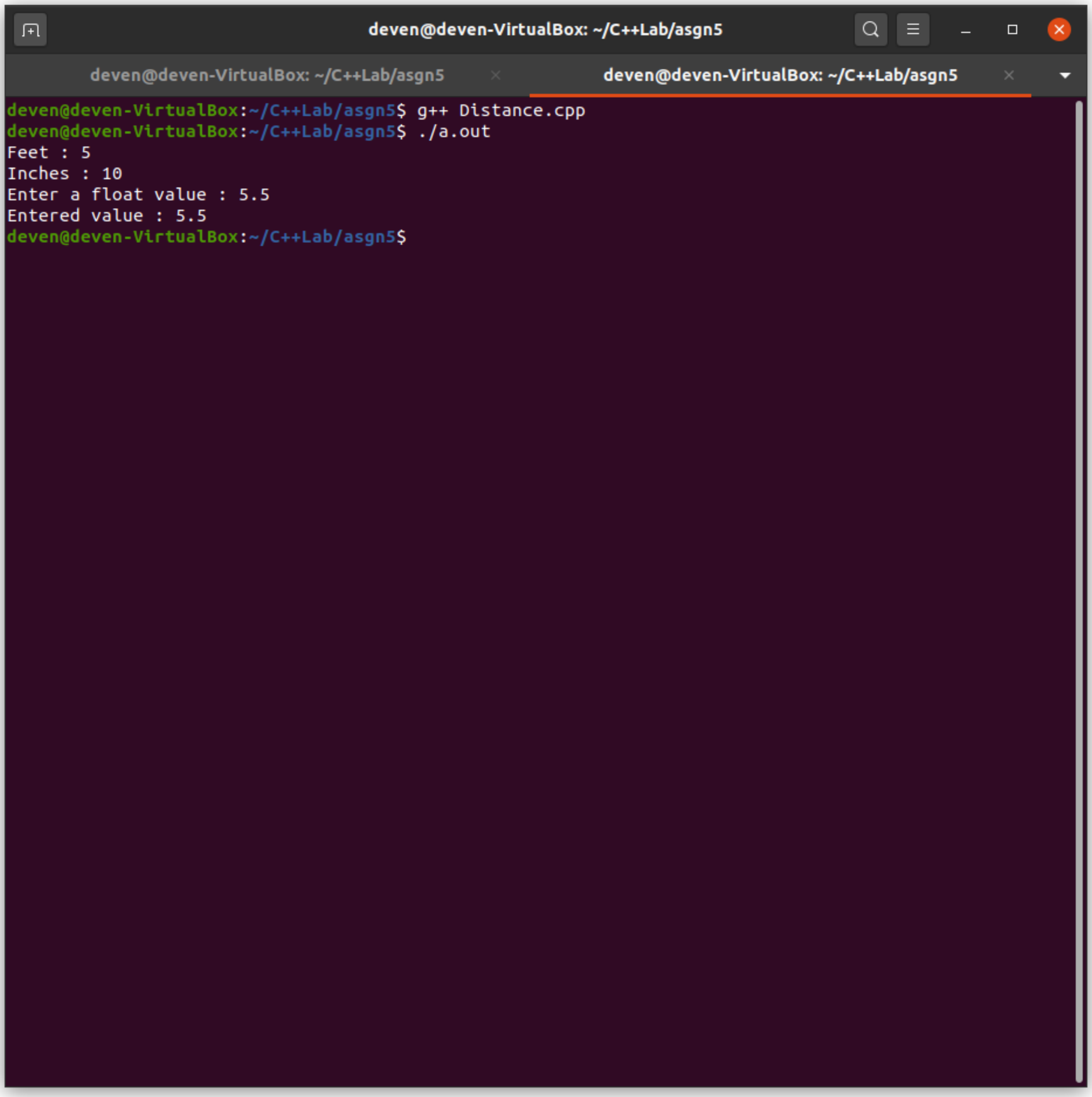
2. Write a C++ program to create a struct Distance with feet and inches. Implement the following overloading functions display

i. display(Dist d) - d is an object of Distance

ii. display(Float f) - f is an input from the user

```
#include <iostream>

using namespace std;


struct distance
```

```
{
	float feet,inches;
};
typedef struct distance Distance;

void display(Distance d)
{
	cout<<"Feet : "<<d.feet<<endl;
	cout<<"Inches : "<<d.inches<<endl;
}

void display(float f)
{
	cout<<"Entered value : "<<f<<endl;
}

int main()
{
	Distance d={5,10};
	display(d);
	float f;
	cout<<"Enter a float value : ";
	cin>>f;
	display(f);
	return 0;
}
```

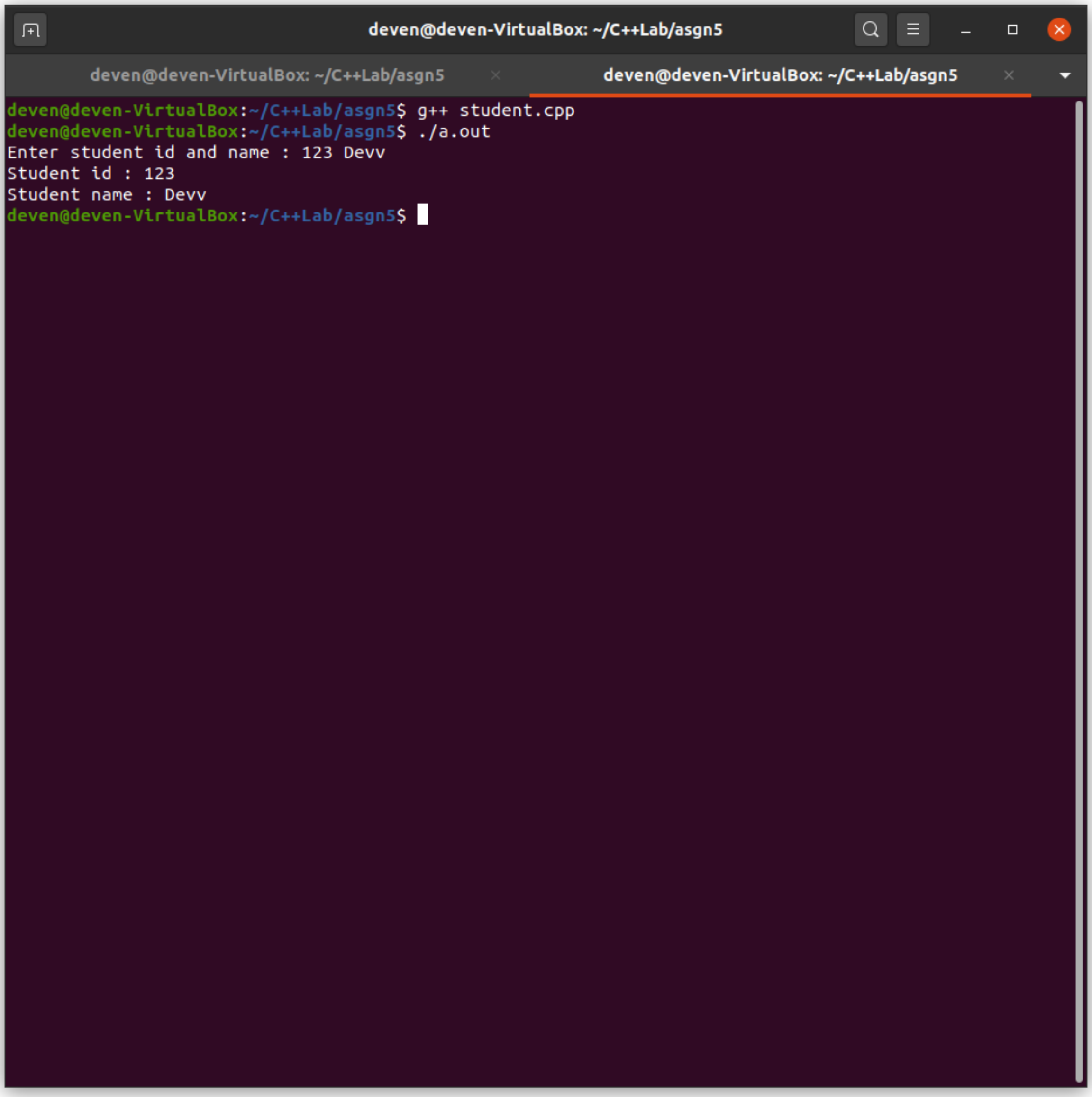

deven@deven-VirtualBox: ~/C++Lab/asgn5
deven@deven-VirtualBox: ~/C++Lab/asgn5
deven@deven-VirtualBox: ~/C++Lab/asgn5

```
deven@deven-VirtualBox:~/C++Lab/asgn5$ g++ Distance.cpp
deven@deven-VirtualBox:~/C++Lab/asgn5$ ./a.out
Feet : 5
Inches : 10
Enter a float value : 5.5
Entered value : 5.5
deven@deven-VirtualBox:~/C++Lab/asgn5$
```

3. Create a namespace student with studentID and name, use a .h file to place this namespace. Display the student details from another file (Student.cpp)

```
//student.h
using namespace std;//for string , std::string name;
namespace student
{
        int id;
```

```
    string name;
}
//student.cpp
#include <iostream>
#include "student.h"
using namespace std;

using student::id;
using student::name;

int main()
{
    cout<<"Enter student id and name : ";
    cin>>id>>name;
    cout<<"Student id : "<<id<<endl;
    cout<<"Student name : "<<name<<endl;
    return 0;
}
```



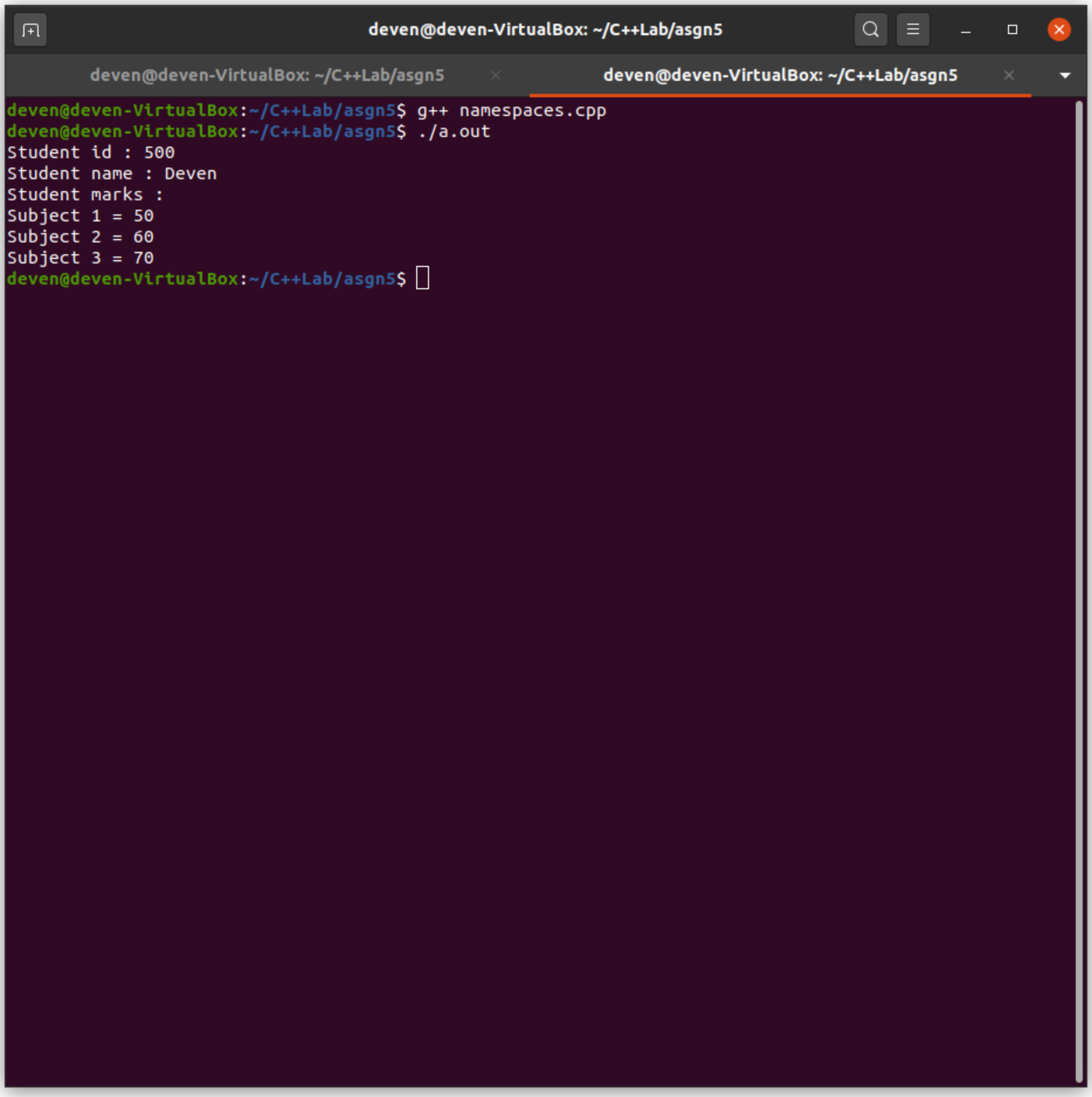
deven@deven-VirtualBox: ~/C++Lab/asgn5
deven@deven-VirtualBox: ~/C++Lab/asgn5
deven@deven-VirtualBox: ~/C++Lab/asgn5

```
deven@deven-VirtualBox:~/C++Lab/asgn5$ g++ student.cpp
deven@deven-VirtualBox:~/C++Lab/asgn5$ ./a.out
Enter student id and name : 123 Devv
Student id : 123
Student name : Devv
deven@deven-VirtualBox:~/C++Lab/asgn5$ 
```

4. Illustrate nesting of namespaces and :: operators suitably.

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

namespace student
{
        int id = 500;
```

```
    string name = "Deven";
    namespace marks
    {
        int subject1 = 50;
        int subject2 = 60;
        int subject3 = 70;
    }
}

int main()
{
    cout<<"Student id : "<<student::id<<endl;
    cout<<"Student name : "<<student::name<<endl;
    cout<<"Student marks : "<<endl;
    cout<<"Subject 1 = "<<student::marks::subject1<<endl;
    cout<<"Subject 2 = "<<student::marks::subject2<<endl;
    cout<<"Subject 3 = "<<student::marks::subject3<<endl;
    return 0;
}
```



deven@deven-VirtualBox: ~/C++Lab/asgn5
deven@deven-VirtualBox: ~/C++Lab/asgn5
deven@deven-VirtualBox: ~/C++Lab/asgn5

```
deven@deven-VirtualBox:~/C++Lab/asgn5$ g++ namespaces.cpp
deven@deven-VirtualBox:~/C++Lab/asgn5$ ./a.out
Student id : 500
Student name : Deven
Student marks :
Subject 1 = 50
Subject 2 = 60
Subject 3 = 70
deven@deven-VirtualBox:~/C++Lab/asgn5$ 
```