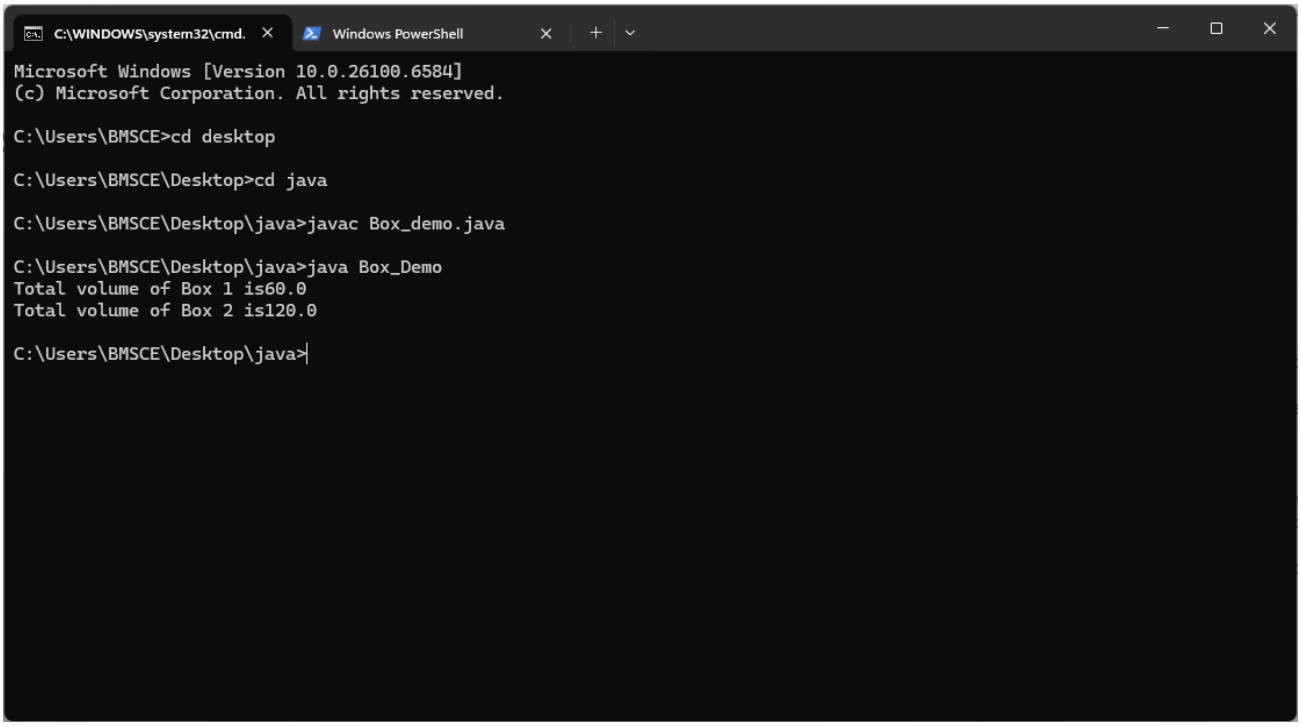
Program 1

```
class Box
{
   double width;
   double height;
   double length;
   Box(double w,double h,double l)
   {
      width = w;
      height = h;
      length = l;
   }
   double volume()
   {
      double volume = width * height * length;
      return volume;
   }
}
class Box_Demo
{
   public static void main(String a[])
   {
      Box B1 = new Box(3,4,5);
      Box B2 = new Box(6,10,2);
      System.out.println("Total volume of Box 1 is  " +  B1.volume());
      System.out.println("Total volume of Box 2 is  " + B2.volume());
   }

}
```

OUTPUT :



```
Microsoft Windows [Version 10.0.26100.6584]
(c) Microsoft Corporation. All rights reserved.

C:\Users\BMSCE>cd desktop

C:\Users\BMSCE\Desktop>cd java

C:\Users\BMSCE\Desktop\java>javac Box_demo.java

C:\Users\BMSCE\Desktop\java>java Box_Demo
Total volume of Box 1 is60.0
Total volume of Box 2 is120.0

C:\Users\BMSCE\Desktop\java>
```

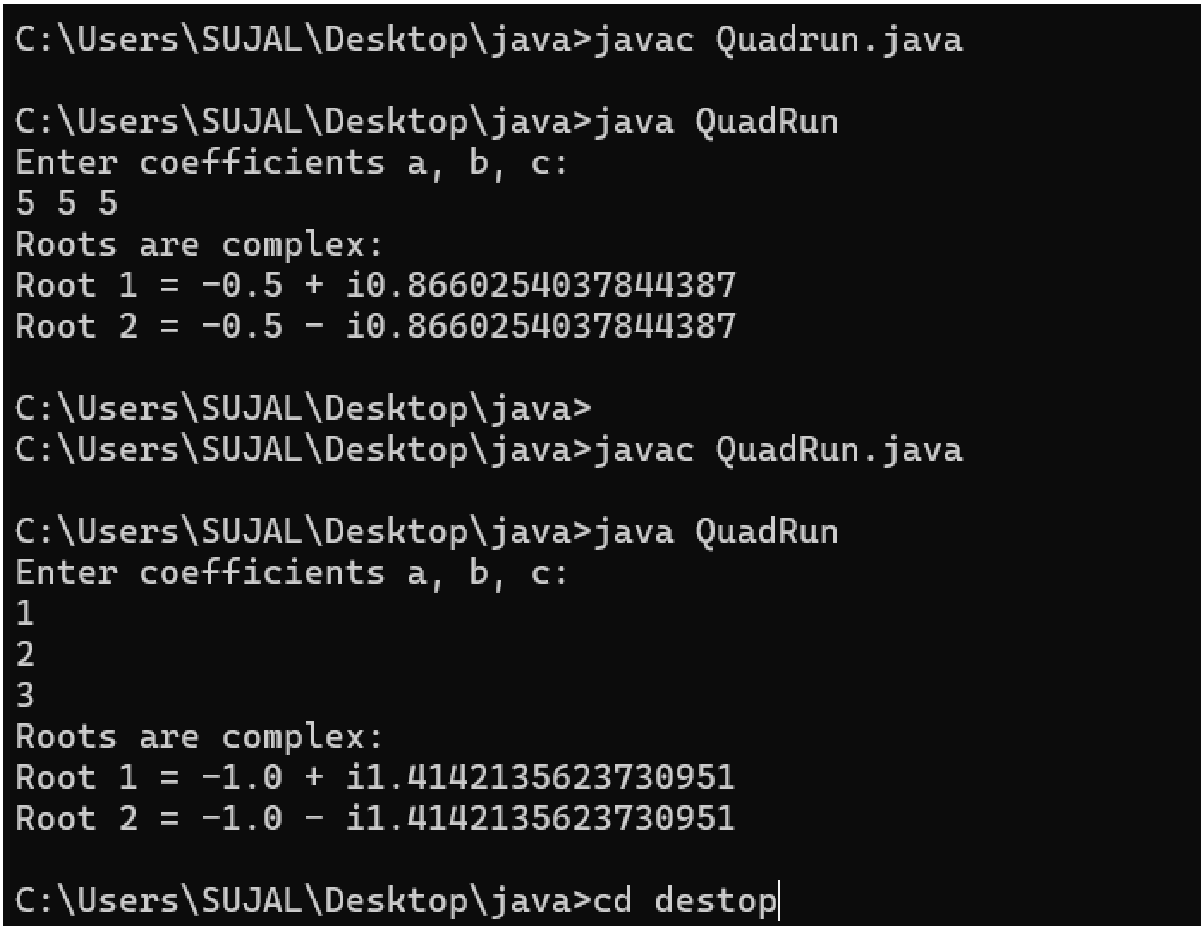
Program 2

```
import java.util.Scanner;
class Quad
{
   double a, b, c;
   double d, root1, root2, real, imag;
   void input()
   {
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter coefficients a, b, c:");
      a = sc.nextDouble();
      b = sc.nextDouble();
      c = sc.nextDouble();
   }
```

```
    void calcRoots()
    {
        d = (b * b) - (4 * a * c);
        if (d == 0)
        {
            root1 = root2 = -b / (2 * a);
            System.out.println("Roots are real and equal: " + root1);
        }
        else if (d > 0)
        {
            root1 = (-b + Math.sqrt(d)) / (2 * a);
            root2 = (-b - Math.sqrt(d)) / (2 * a);
            System.out.println("Roots are real and distinct:");
            System.out.println("Root 1 = " + root1);
            System.out.println("Root 2 = " + root2);
        }
         else {
            real = -b / (2 * a);
            imag = Math.sqrt(-d) / (2 * a);
            System.out.println("Roots are complex:");
            System.out.println("Root 1 = " + real + " + i" + imag);
            System.out.println("Root 2 = " + real + " - i" + imag);
        }
    }
}
 class QuadRun
```

```
{
  public static void main(String[] args)
  {
     Quad q = new Quad();
     q.input();
     q.calcRoots();
  }
}
```

OUTPUT:



```
C:\Users\SUJAL\Desktop\java>javac Quadrun.java

C:\Users\SUJAL\Desktop\java>java QuadRun
Enter coefficients a, b, c:
5 5 5
Roots are complex:
Root 1 = -0.5 + i0.8660254037844387
Root 2 = -0.5 - i0.8660254037844387

C:\Users\SUJAL\Desktop\java>
C:\Users\SUJAL\Desktop\java>javac QuadRun.java

C:\Users\SUJAL\Desktop\java>java QuadRun
Enter coefficients a, b, c:
1
2
3
Roots are complex:
Root 1 = -1.0 + i1.4142135623730951
Root 2 = -1.0 - i1.4142135623730951

C:\Users\SUJAL\Desktop\java>cd destop
```