

Experiment 2.1.1

Roots of an experiment

Algorithm : Step 1 : Start.

Step 2 : Read three space-separated integers a, b, and c.

Step 3 : Calculate the discriminant.

$D = b^2 - 4ac$

Step 4 : Check the value of the discriminant .

- Case1: If $D > 0$ (Real and different roots)

Print: root1,root2.

- Case 2: If $D == 0$ (Real and same roots)

Print : root1=root2.

- Case 3: If $D < 0$ (Imaginary roots)

Print : Root = root 1 + imaginary

Root = root 2 + imaginary Step 5 :

Stop.

Code:

```
a, b, c = map(float, input().split())
```

```
D = (b*b) - (4*a*c) sqrd
```

```
= D ** 0.5
```

```
if D > 0:
```

```
    root1 = (-b + sqrd) / (2*a)
```

```
    root2 = (-b - sqrd) / (2*a)
```

```
    print("root1 = " f"{root1:.2f}")
```

```
    print("root2 = " f"{root2:.2f}") elif D
```

```
== 0:
```

$$\text{root1} = \text{root2} = -b / (2*a)$$

```
print("root1 = root2 = " f"\{root1:.2f}")
```

else:

```
    real = (-b) / (2*a)    imaginary = sqrtD /
(2*a)    print(f"root1 =
{real:.2f}+{imaginary:.2f}i")    print(f"root2 =
{real:.2f}-{imaginary:.2f}i")
```

The screenshot shows the CodeDETTANTRA IDE interface with the following details:

- Title Bar:** CodeDETTANTRA Home
- Code Editor:**

```
2.1.1 Roots of a Quadratic Equation
Write a program to find the roots of a quadratic equation, given its coefficients a, b, and c. Use the quadratic formula:  $\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ 
The discriminant  $D = b^2 - 4ac$  determines the nature of the roots.
• If  $D > 0$ : Roots are real and different
• If  $D = 0$ : Roots are real and the same
• If  $D < 0$ : Roots are imaginary
```
- Input Format:**
 - Three space-separated integers representing the coefficients a, b, and c, respectively.
 - If roots are real and different, print:
- Output Format:**
 - If roots are the same, print:
- Code Content:**

```
1 a, b, c = map(int, input().split())
2 d = b**2 - 4*a*c
3 sqrd = d ** 0.5
4 if d > 0:
5     root1 = (-b + sqrd) / (2*a)
6     root2 = (-b - sqrd) / (2*a)
7     print(f"root1 = {root1:.2f}i")
8     print(f"root2 = {root2:.2f}i")
9 elif d == 0:
10    root = -b / (2*a)
11    print(f"root1 = root2 = {root:.2f}i")
```
- Test Cases:**
 - Test Case 1:** Average time 0.006 s, Maximum time 0.010 s, 3 out of 3 shown test case(s) passed, 3 out of 3 hidden test case(s) passed.
 - Test Case 2:** Average time 6.50 ms, Maximum time 10.00 ms, 3 out of 3 shown test case(s) passed, 3 out of 3 hidden test case(s) passed.
- Bottom Navigation:** < Prev, Reset, Submit, Next >

FlowChart:

S

