

# EXPERIMENT - 2

Name : Om Kashikar

PRN : 25070521170

## 2.1.1 Roots of Quadratic Equation

### ALGORITHM

1. Start
2. Import the math library.
3. Read three integers a, b, and c (coefficients of the quadratic equation).
4. Calculate the discriminant

$$D = b^2 - 4ac$$

5. If  $D > 0$ :

Calculate two real and different roots using:

$$\frac{-b+\sqrt{D}}{2a}, \frac{-b-\sqrt{D}}{2a}$$

Print both roots up to 2 decimal places.

- 6 Else if  $D == 0$ :

Calculate the single repeated root:

$$\frac{-b}{2a}$$

Print the root twice up to 2 decimal places.

- 7 Else ( $D < 0$ ):

Calculate real part:

$$\frac{-b}{2a}$$

Calculate imaginary part:

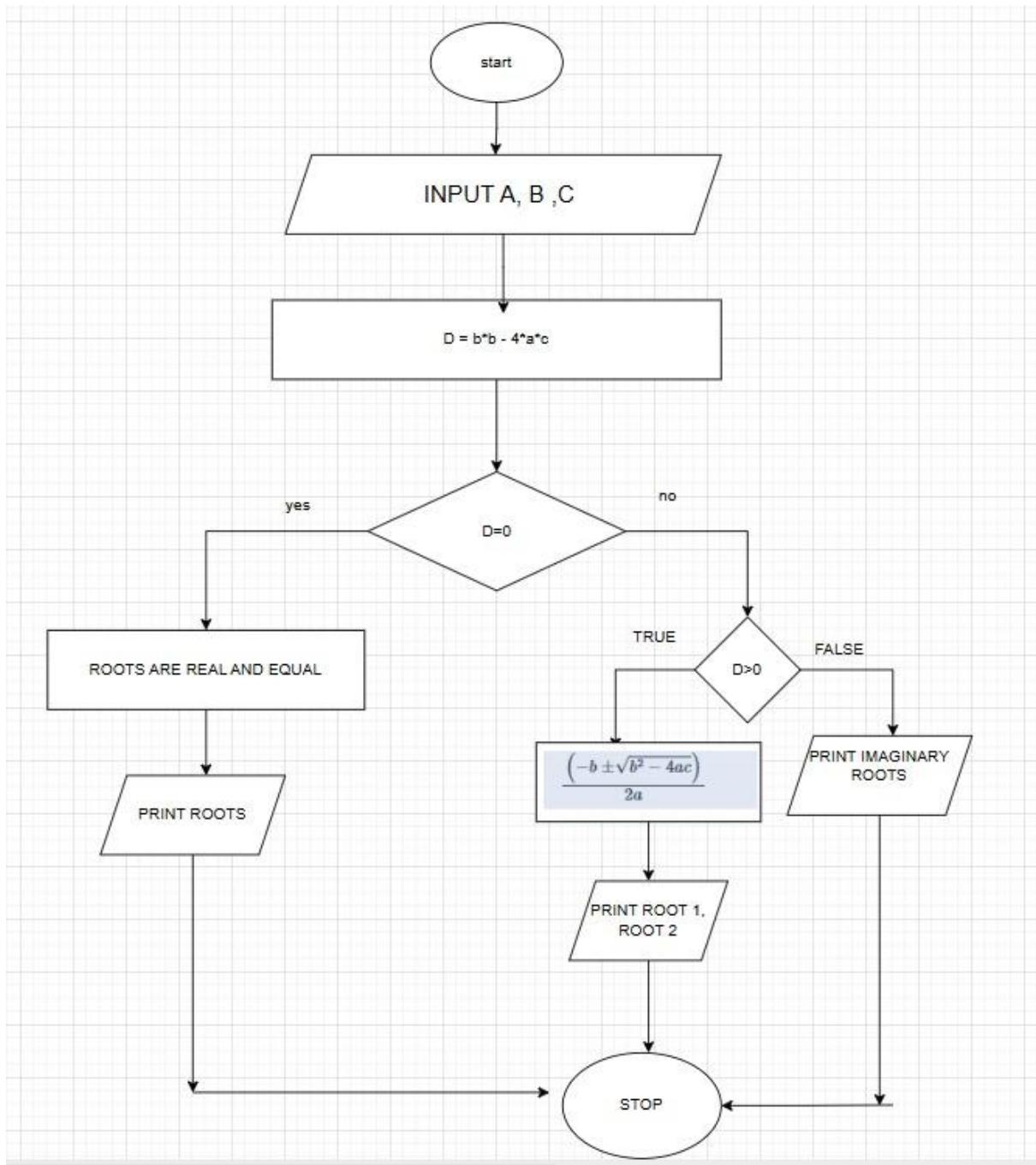
$$\frac{\sqrt{-D}}{2a}$$

Print both complex roots up to 2 decimal places.

- 8 Stop

### FLOWCHART

# EXPERIMENT - 2



## PYTHON CODE

```
import math
```

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

```
D = b*b - 4*a*c
```

# EXPERIMENT - 2

if  $D > 0$ :

```
root1 = (-b + math.sqrt(D)) / (2*a)
root2 = (-b - math.sqrt(D)) / (2*a)
print(f"root1 = {root1:.2f}")
print(f"root2 = {root2:.2f}")
```

elif  $D == 0$ :

```
root = (-b) / (2*a)
print(f"root1 = root2 = {root:.2f}")
```

else:

```
real = (-b) / (2*a)
imag = math.sqrt(-D) / (2*a)
print(f"root1 = {real:.2f}+{imag:.2f}i")
print(f"root2 = {real:.2f}-{imag:.2f}i")
```

## EXECUTION

The screenshot shows the CodeTantra IDE interface. The top bar displays the user information: om.kashikar.batch2025@sitnagpur.siu.edu.in, Support, and Logout.

The left sidebar shows the problem statement: "2.1.1. Roots of a Quadratic Equation". It includes instructions to find roots of a quadratic equation given coefficients  $a$ ,  $b$ , and  $c$  using the formula  $\frac{(-b \pm \sqrt{b^2 - 4ac})}{2a}$ . It also notes that 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 same; if  $D < 0$ , roots are imaginary.

The right panel shows the code editor with the following Python code:

```
quadratic...
import math
a, b, c = map(int, input().split())
D = b*b - 4*a*c
if D > 0:
    root1 = (-b + math.sqrt(D)) / (2*a)
    root2 = (-b - math.sqrt(D)) / (2*a)
    print(f"root1 = {root1:.2f}")
    print(f"root2 = {root2:.2f}")
else:
    real = (-b) / (2*a)
    imag = math.sqrt(-D) / (2*a)
    print(f"root1 = {real:.2f}+{imag:.2f}i")
    print(f"root2 = {real:.2f}-{imag:.2f}i")
```

The code editor includes performance metrics: Average time 0.006 s, Maximum time 0.008 s, and 6.50 ms. It also shows test results: 3 out of 3 shown test case(s) passed and 3 out of 3 hidden test case(s) passed.

The bottom section shows sample test cases and terminal output. The terminal output for the provided test case (1 -5 6) shows the expected output as 1 -5 6 and the actual output as 1 -5 6, root1 = -3.00, and root2 = -2.00.