

2.1.1 Roots of a Quadratic Equation:

Algorithm:

Step 1: Start

Step 2: Input values of a, b and c

Step 3: Calculate the discriminant

$$D = b^2 - 4*a*c$$

Step 4: if $D > 0$, then

 Calculate two real and different roots

 Display root1 and root2

Step 5: Else if $D == 0$, then

 Calculate one real and equal root

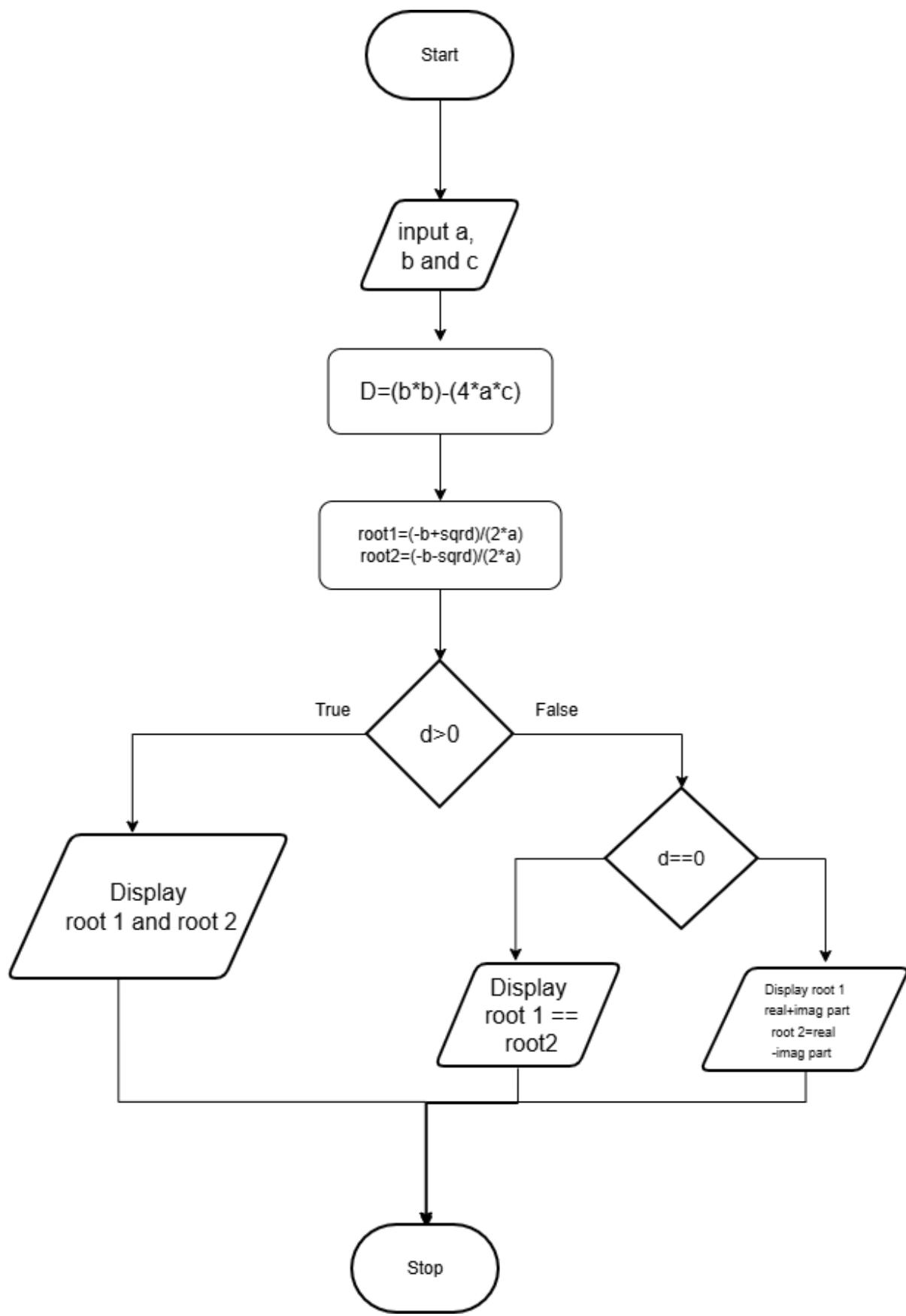
Step 6: Else ($D < 0$) , then

 Calculate two complex roots

 Display complex root1 and root2

Step 7: Stop

Flowchart:



2.1. Roots of a Quadratic Equation

38:00     -

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.

Output Format:

- If roots are real and different, print:

```
root1 = <Root1>
root2 = <Root2>
```

- If roots are the same, print:

```
root1 = root2 = <Root1>
```

- If roots are imaginary, print:

```
root1 = <RealPart>+<ImaginaryPart>i
root2 = <RealPart>-<ImaginaryPart>i
```

- All values should be formatted to two decimal places.

Sample Test Cases

Explorer
 quadratic...

```

1 a, b, c=map(float,input().split())
2 d=(b*b)-4*a*c
3 sqrd=d**0.5
4 root1=(-b-sqrd)/(2*a)
5 root2=(-b+sqrd)/(2*a)
6 v.if(d>0):
7   →print(f"root1 = {root1:.2f}")
8   →print(f"root2 = {root2:.2f}")
9 v.elif(d==0):
10  →print(f"root1 = root2 = {root1:.2f}")
11 v.else:
12   →print(f"root1 = {root1.real:.2f}+{root1.imag:.2f}i")
13   →print(f"root2 = {root2.real:.2f}{root2.imag:.2f}i")
```

Average time
0.015 s
Maximum time
0.025 s

14.67 ms
25.00 ms

 3 out of 3 shown test case(s) passed
 3 out of 3 hidden test case(s) passed

 Test case 1 **25 ms**
 Expected output

Actual output
   ^

1 -5 6
1 -5 6

root1 = -3.00
root1 = -3.00

root2 = -2.00
root2 = -2.00

 Test case 2 **14 ms**
 Expected output

Actual output
   ^

1 -4 4
1 -4 4

root1 = -root2 = -2.00
root1 = -root2 = -2.00

 Terminal
 Test cases