

## Experiment 01:-

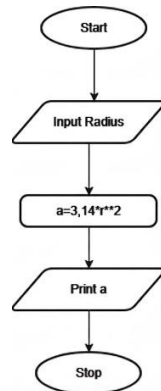
### Problem Statement:-

Write a Python program that calculates the area of a circle when the radius is provided by the user. Use  $\pi = 3.14$  and display the area.

### Algorithm:-

1. Start
2. Read the radius  $r$  from the user
3. Calculate the area using the formula:  
 $\text{Area} = 3.14 \times r \times r$
4. Display the area
5. Stop.

### Flowchart:-



### Execution:-

The screenshot shows the CodeTANTRA IDE interface. On the left, the problem statement is displayed: "Write a Python program that calculates the area of a circle when the radius is provided by the user. Use  $\pi = 3.14$  and display the area." Below this, the input and output formats are specified. The input format is "A single line containing a floating-point number representing the radius." The output format is "Print the computed area of the circle formatted to 4 decimal places." The sample test cases section is also visible.

On the right, the Python code is shown in the editor:

```
radius = float(input())
area = 3.14 * radius * radius
print(f"{area:.4f}")
```

Below the code, the execution results are displayed. The program executed successfully, and the output is shown as 35.4403. The test cases section indicates that 2 out of 2 shown test case(s) passed and 2 out of 2 hidden test case(s) passed.

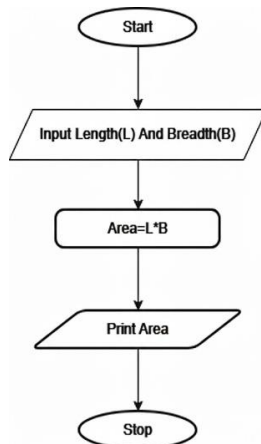
## Problem Statement:-

Write a Python program to calculate the area of a rectangle given its length and width.

## Algorithm:-

1. Start.
2. Read the length l and Width b from the user.
3. Calculate the area using the formula:  
 $\text{Area} = l * b$ .
4. Display the area.
5. Stop.

## Flowchart:-



## Execution:-

**CODETANTRA** Home

1.2 Area of Rectangle

Write a Python program to calculate the area of a rectangle given its length and width.

**Formula:**  
Area of Rectangle = Length × Width

**Input Format:**

- First line contains a float value representing the length of the rectangle
- Second line contains a float value representing the width of the rectangle

**Output Format:**

- Print the area of the rectangle as a float value formatted to 2 decimal places.

```

1 length = float(input())
2 width = float(input())
3 area = length * width
4 print(f"{area:.2f}")
  
```

Test Results:

Test Case	Expected Output	Actual Output	Status
Test case 1	38.50	38.50	Passed
Test case 2	54.00	54.00	Passed

Summary: 5 out of 5 shown test case(s) passed, 5 out of 5 hidden test case(s) passed.

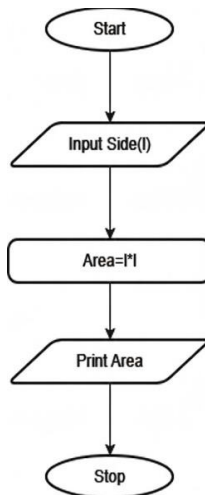
Buttons: Prev, Next, Submit, Test Cases

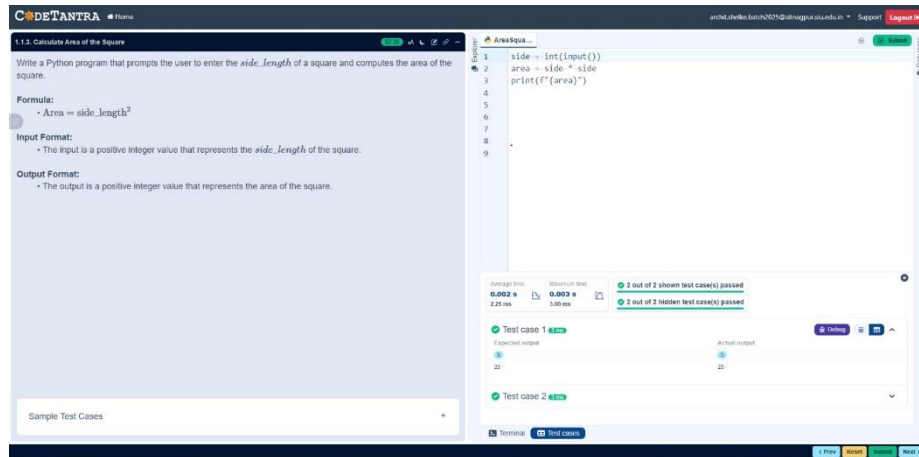
**Problem Statement:-**

Write a Python program that prompts the user to enter the Side Length of a square and computes the area of the square.

**Algorithm:-**

1. Start.
2. Read the side length  $s$  of the square.
3. Calculate the area using the formula:  $\text{Area}=s \times s$ .
4. Display the area
5. Stop.

**Flowchart:-****Execution:-**



3

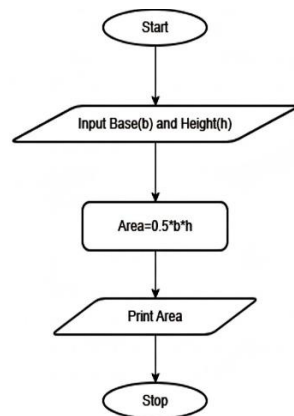
### Problem Statement:-

Write a Python program that prompts the user to enter the triangle's base and height and computes the triangle's area.

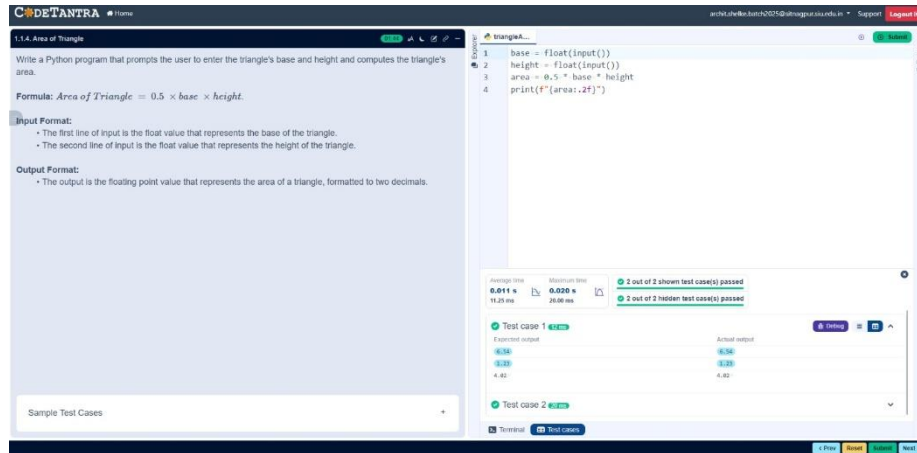
### Algorithm:-

1. Start.
2. Read the base  $b$  and height  $h$  of the triangle.
3. Calculate the area using the formula:  $\text{Area} = 0.5 \times b \times h$ .
4. Display the area.
5. Stop.

### Flowchart:-



### Execution:-



4

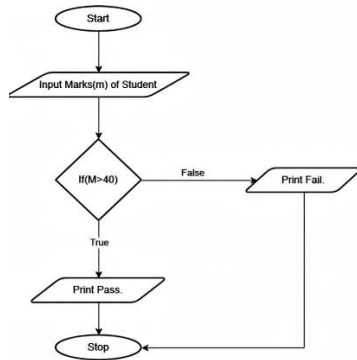
### Problem Statement:-

Write a Python program to determine whether a student passed the exam or not based on their marks.

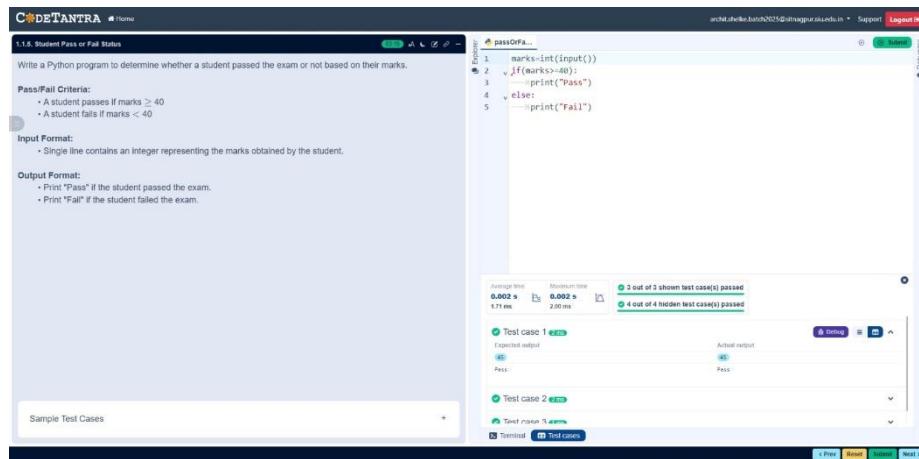
### Algorithm:-

1. Start.
2. Read the marks obtained by the student.
3. If marks > 40 → Display “Pass”.
4. Else → Display “Fail”.
5. Stop.

## Flowchart:-



## Execution:-



## Experiment 02:-

### Problem Statement:-

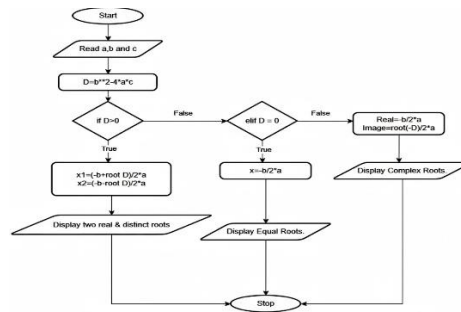
Write a program to find the roots of a quadratic equation, given its coefficients a, b, and c. Use the quadratic formula:  $(-b \pm \sqrt{b^2 - 4ac}) / 2a$ .

### Algorithm:-

1. Start.
2. Read the coefficients a, b, and c.
3. Calculate the discriminant using:-  $D = b^2 - 4ac$ .

4. If  $D > 0$  Compute:-  $x_1 = \frac{-b + \sqrt{D}}{2a}$  and  $x_2 = \frac{-b - \sqrt{D}}{2a}$  Display two real and distinct roots.
5. Else if  $D = 0$  Compute:-  $x = -b/2a$  Display equal real roots.
6. Else ( $D < 0$ ) Compute:- Real part =  $-b/2a$  Imaginary part =  $\sqrt{-D}/2a$  Display complex roots.
7. Stop.

### Flowchart:-



### Execution:-

```

1 import math
2 a, b, c = map(int, input().split())
3 D = b*b - 4*a*c
4
5 if D > 0:
6     root1 = (-b + math.sqrt(D)) / (2*a)
7     root2 = (-b - math.sqrt(D)) / (2*a)
8     print("root1 = {root1:.2f}"
9         "root2 = {root2:.2f}")
10
11 elif D == 0:
12     root = -b / (2*a)
13     print("root1 = root2 = {root:.2f}")
14
15 else:
16     real_part = -b / (2*a)
17     imaginary_part = math.sqrt(-D) / (2*a)
18     print("root1 = {real_part:.2f}+{imaginary_part:.2f}i"
19         "root2 = {real_part:.2f}-{imaginary_part:.2f}i")
20
21
22
23
24
25
26
  
```

Test Case 1: 3.000 3.000 3.000  
Expected output: root1 = 3.00 root2 = 3.00  
Actual output: root1 = 3.00 root2 = 3.00

Test Case 2: 1.000 2.000 2.000  
Expected output: root1 = 1.00 root2 = 2.00  
Actual output: root1 = 1.00 root2 = 2.00

## Experiment 03:-

### Problem Statement:-

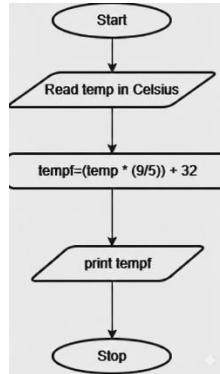
Write a Python program to convert temperature from Celsius to Fahrenheit.

### Algorithm:-

1. Start.
2. Input temperature in Celsius C.
3. Calculate Fahrenheit using the formula  $F = (C \times 9/5) + 32$ .

4. Display the temperature in Fahrenheit.
5. Stop.

### Flowchart:-



### Execution:-

The screenshot shows the CodeTANTRA web interface. On the left, the problem statement for '5.1.2 Celsius to Fahrenheit' is displayed, including the formula  $Fahrenheit = (Celsius \times \frac{9}{5}) + 32$  and input/output specifications. The main editor on the right contains the following Python code:

```
1 celsius = float(input())
2 fahrenheit = (celsius * 9/5) + 32
3
4 print(f'{fahrenheit:.2f}')
```

Below the code editor, the execution results are shown. The program ran successfully, with a runtime of 0.002s and a memory usage of 0.003s. The test results indicate that 4 out of 4 shown test cases passed and 4 out of 4 hidden test cases passed. A table of test cases is provided:

Test Case	Expected output	Actual output
Test case 1	32.00	32.00
Test case 2	32.00	32.00

The interface also includes a 'Sample Test Cases' input field and buttons for 'Print', 'Submit', 'Solve', and 'Next'.

### Problem Statement:-

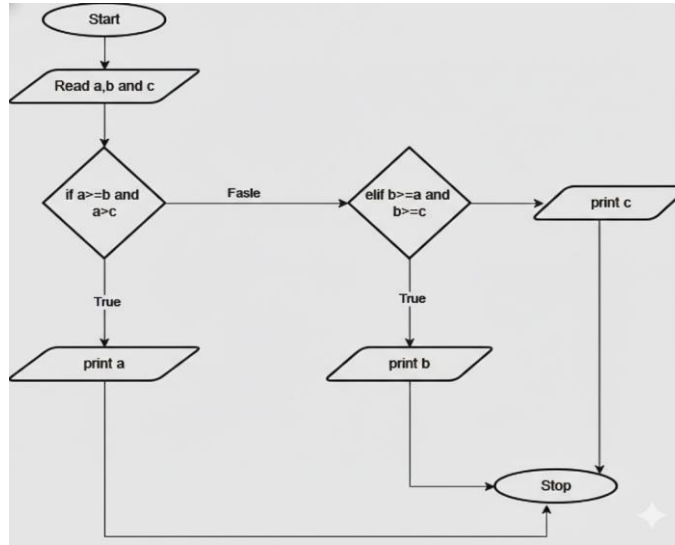
Write a Python program that prompts the user to enter three integers. Print the largest of the three integers.



## Algorithm:-

1. Start.
2. Input three integers a, b, and c.
3. If  $a \geq b$  and  $a \geq c \rightarrow$  Print a as the largest number.
4. Else if  $b \geq a$  and  $b \geq c \rightarrow$  Print b as the largest number.
5. Else  $\rightarrow$  Print c as the largest number.
6. Stop

## Flowchart:-



## Execution:-

The screenshot shows the CodeTANTRA IDE interface. On the left, the file explorer shows '5.1.1 Largest of Three Numbers'. The main editor displays a Python program:

```
1 a = int(input())
2 b = int(input())
3 c = int(input())
4
5 print(max(a, b, c))
```

Below the code editor, the test results are shown. It indicates that 2 out of 2 shown test cases passed and 2 out of 2 hidden test cases passed. The test case details show the expected output and actual output for a given input.

Test Case	Expected output	Actual output
Test case 1	7	7

## Experiment 04:-

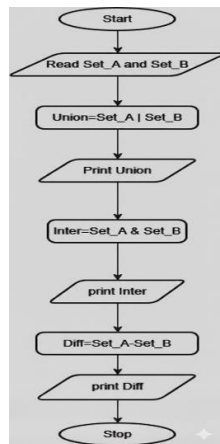
### Problem Statement:-

Write a Python program to perform union, intersection and difference operations on Set A and Set B.

### Algorithm:-

1. Start.
2. Declare two sets A and B
3. Read elements of Set A from the user
4. Read elements of Set B from the user
5. Perform Union operation  $\text{Union} = A \cup B$ .
6. Perform Intersection operation  $\text{Intersection} = A \cap B$ .
7. Perform Difference operations
8.  $\text{Difference1} = A - B$ .
9. Display Set A and Set B
10. Display Union, Intersection, and Difference results
11. Stop.

### Flowchart:-



### Execution:-

The screenshot displays the CodeTANTRA IDE interface. On the left, the problem statement and input/output formats are visible. The main editor shows the Python code for set operations. The output pane on the right shows the results of the program execution for two test cases.

**CodeTANTRA** Home

4.1.1. Set Operations

Write a Python program to perform union, intersection and difference operations on Set A and Set B.

**Input Format:**

- First Line prompts "Set A:" followed by space-separated list of integers for Set A.
- The second input prompts "Set B:" followed by space-separated list of integers for Set B.

**Output Format:**

- The first line prints "Union:" followed by the union of Set A and Set B.
- The second line prints "Intersection:" followed by the Intersection of Set A and Set B.
- The third line prints "Difference:" followed by the difference of Set A and Set B.

**Note:**

- If there is no intersection between the two sets, the program prints an empty set, which appears as "set()" in the output.
- Please refer to the visible test cases for better understanding.

Sample Test Cases

```
1 setoperet...
2 set_a = set(map(int, input("Set A: ").split()))
3 set_b = set(map(int, input("Set B: ").split()))
4
5 union_set = set_a | set_b
6 intersection_set = set_a & set_b
7 difference_set = set_a - set_b
8
9 print("Union:", union_set)
10 print("Intersection:", intersection_set)
11 print("Difference:", difference_set)
12
13
```

Average Time: 0.003 s, 3.25 ms | Maximum Time: 0.004 s, 4.00 ms | 2 out of 2 shown test case(s) passed | 2 out of 2 hidden test case(s) passed

Test case 1

Expected output	Actual output
Set A: 1 2 3 4 5 6	Set A: 1 2 3 4 5 6
Set B: 2 3 4 5 6 7	Set B: 2 3 4 5 6 7
Union: {1, 2, 3, 4, 5, 6, 7}	Union: {1, 2, 3, 4, 5, 6, 7}
Intersection: {2, 3, 4, 5, 6}	Intersection: {2, 3, 4, 5, 6}
Difference: {1, 7}	Difference: {1, 7}

Terminal | Test cases

Run | Stop | Submit | Next

