

Experiment – 1.1.1

1 Area of Circle

- Algorithm

STEP 1 : Start

STEP 2 : Input radius

STEP 3 : Calculate

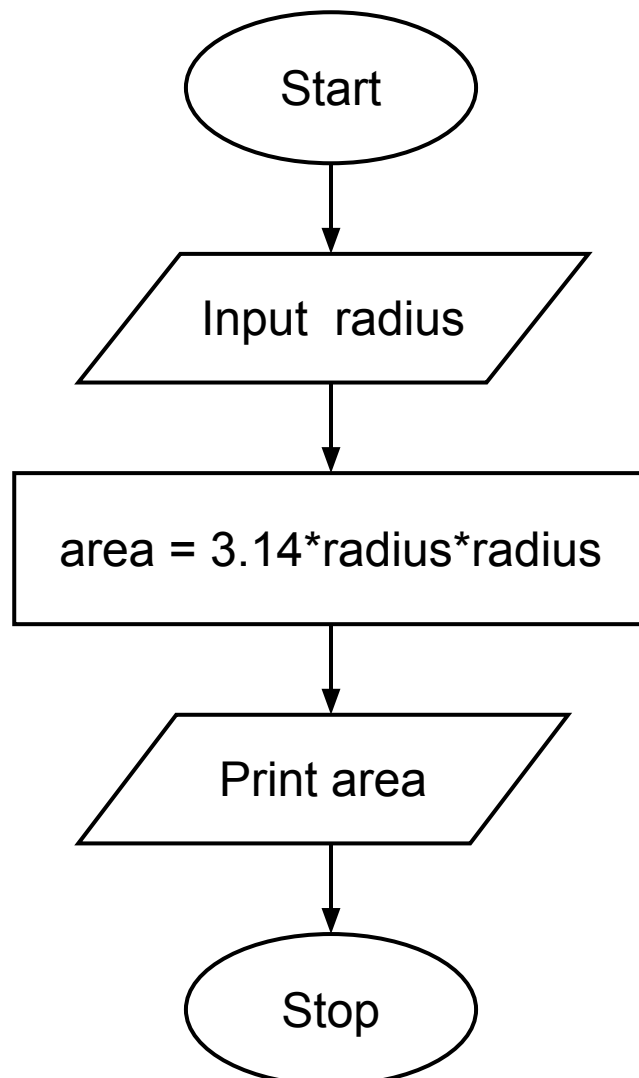
area =

$3.14 * \text{radius} * \text{radius}$

STEP 4 : Print area

STEP 5 : Stop

- Flowchart



- Code

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

The screenshot displays the CodeTANTRA web interface. The top navigation bar includes the CodeTANTRA logo, a home icon, the user email 'vedant.awachat.batch2025@sitnagpur.siu.edu.in', and links for 'Support' and 'Logout'. The main content area is titled '1.1.1. Area of Circle' and contains the following instructions:

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.

Input Format:

- A single line containing a floating-point number representing the radius.

Output Format:

- Print the computed area of the circle formatted to 4 decimal places.

Below the instructions is a 'Sample Test Cases' section with a '+' icon. The right-hand side of the interface shows a code editor with the following Python code:

```
1 #Write your code here...
2 radius = float(input())
3 area = 3.14*radius*radius
4 print(f"{area:.4f}")
```

Below the code editor, the execution results are displayed:

Average time	Maximum time	Test Results
0.017 s 16.75 ms	0.049 s 49.00 ms	2 out of 2 shown test case(s) passed 2 out of 2 hidden test case(s) passed

Below the table, 'Test case 1' is expanded, showing the following details:

- Test case 1: 49 ms
- Expected output: 35.4493
- Actual output: 35.4493

At the bottom of the interface, there is a 'Terminal' tab and a 'Test cases' tab. The bottom navigation bar includes buttons for '< PREV', 'RESET', 'SUBMIT', and 'NEXT >'.

Experiment – 1.1.2

Area of Rectangle

- Algorithm

STEP 1 : Start

STEP 2 : Input length,
breath

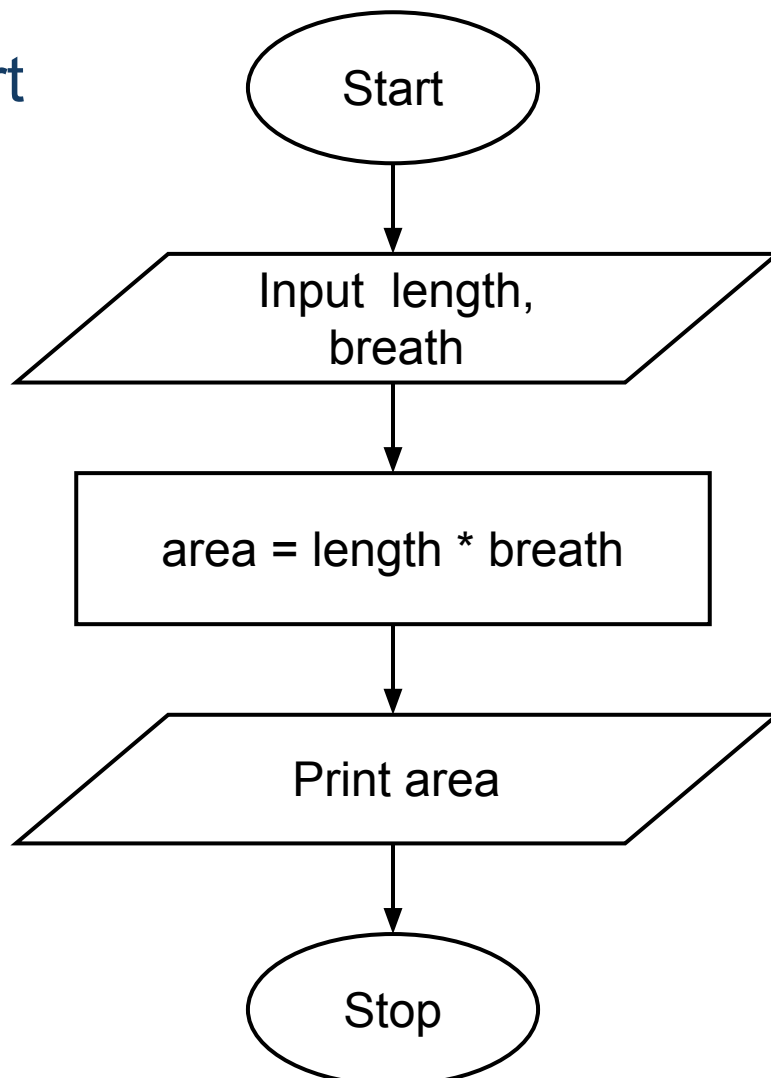
STEP 3 : Calculate
area =

length*breath

STEP 4 : Print area

STEP 5 : Stop

- Flowchart



• Code

```
length = float(input())  
breath = float(input())  
area = length*breath  
print(f"{area:.2f}")
```

CODETANTRA Home

1.1.2. Area of Rectangle 04:31

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.

Sample Test Cases +

areaOfRe...

```
1 # -Type-Content-Here...  
2 length = float(input())  
3 width = float(input())  
4 area = length*width  
5 print(f"{area:.2f}")
```

Average time: 0.009 s (8.70 ms) Maximum time: 0.015 s (15.00 ms)

5 out of 5 shown test case(s) passed
5 out of 5 hidden test case(s) passed

Test case 1 15 ms

Expected output
18.5
54.60

Actual output
18.5
54.60

Terminal Test cases

< PREV RESET SUBMIT NEXT >

Experiment – 1.1.3

Calculate Area of the Square

- Algorithm

STEP 1 : Start

STEP 2 : Input side_length

STEP 3 : Calculate

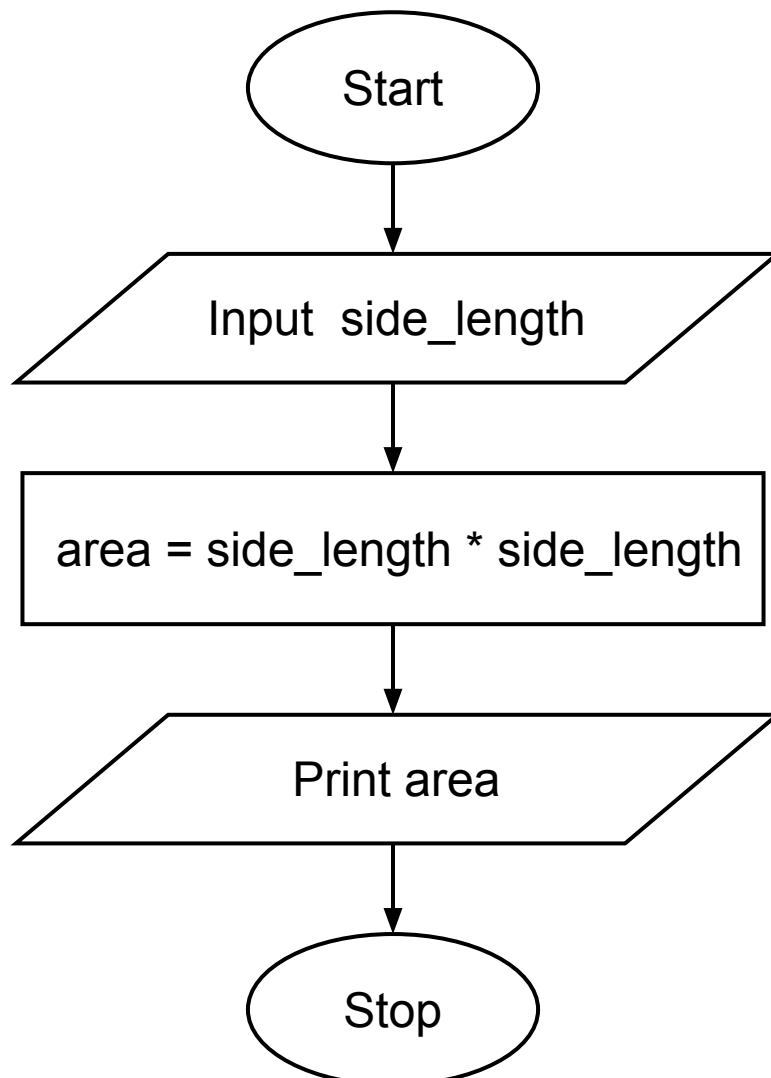
$\text{area} = \text{side_length} * \text{side_length}$

side_length

STEP 4 : Print area

STEP 5 : Stop

- Flowchart



- Code

```
side_length=int(input())
area=side_length*side_length
print(area)
```

CODETANTRA Home vedant.awachat.batch2025@sitnagpur.siu.edu.in Support Logout

1.1.3. Calculate Area of the Square

02:09

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

Formula:

- Area = side_length^2

Input Format:

- The input is a positive integer value that represents the *side_length* of the square.

Output Format:

- The output is a positive integer value that represents the area of the square.

Sample Test Cases

AreaSqua...

```
1 # Write your code here...
2 side_length = int(input())
3 area = side_length * side_length
4 print(area)
5
6
7
8
9
```

Average time: 0.006 s
Maximum time: 0.010 s
5.75 ms 10.00 ms

2 out of 2 shown test case(s) passed
2 out of 2 hidden test case(s) passed

Test case 1 10 ms

Expected output: 25

Actual output: 25

Terminal Test cases

PREV RESET SUBMIT NEXT

Experiment – 1.1.4

Area of Triangle

- Algorithm

STEP 1 : Start

STEP 2 : Input base, height

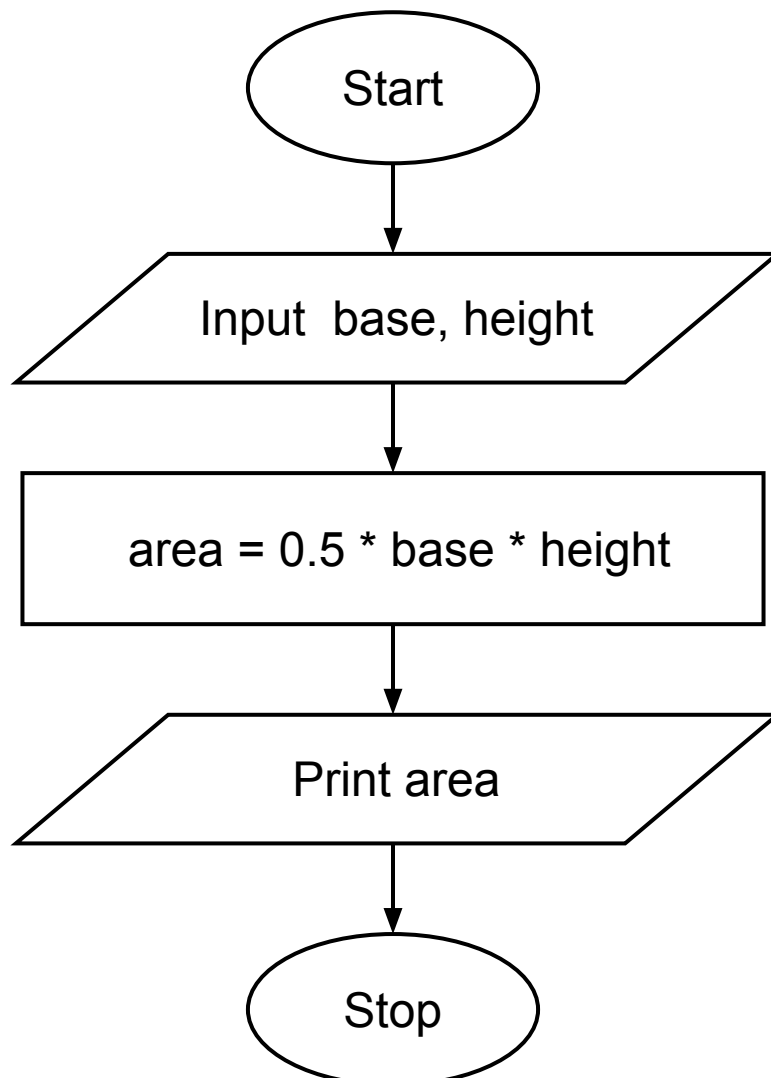
STEP 3 : Calculate

$$\text{area} = 0.5 * \text{base} * \text{height}$$

STEP 4 : Print area

STEP 5 : Stop

- Flowchart



• Code

```
base=float(input())
height=float(input())
area=0.5*base*height
print(f"{area:.2f}")
```

CODETANTRA Home

vedant.awachat.batch2025@sitnagpur.siu.edu.in Support Logout

1.1.4. Area of Triangle

04:20

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

Formula: $Area\ of\ Triangle = 0.5 \times base \times height$.

Input Format:

- The first line of input is the float value that represents the base of the triangle.
- The second line of input is the float value that represents the height of the triangle.

Output Format:

- The output is the floating point value that represents the area of a triangle, formatted to two decimals.

Sample Test Cases

```
1 #-write-your-code-here...
2 base = float(input())
3 height = float(input())
4 area = 0.5*base*height
5 print(f"{area:.2f}")
```

Average time: 0.007 s, 7.50 ms | Maximum time: 0.009 s, 9.00 ms

2 out of 2 shown test case(s) passed
2 out of 2 hidden test case(s) passed

Test case 1 0 ms

Expected output

6.54
1.23

Actual output

6.54
1.23

Terminal Test cases

PREV RESET SUBMIT NEXT

Experiment – 1.1.5

Student Pass or Fail status

- Algorithm

STEP 1 : Start

STEP 2 : Input marks

STEP 3 : Check
condition

If marks \geq

40

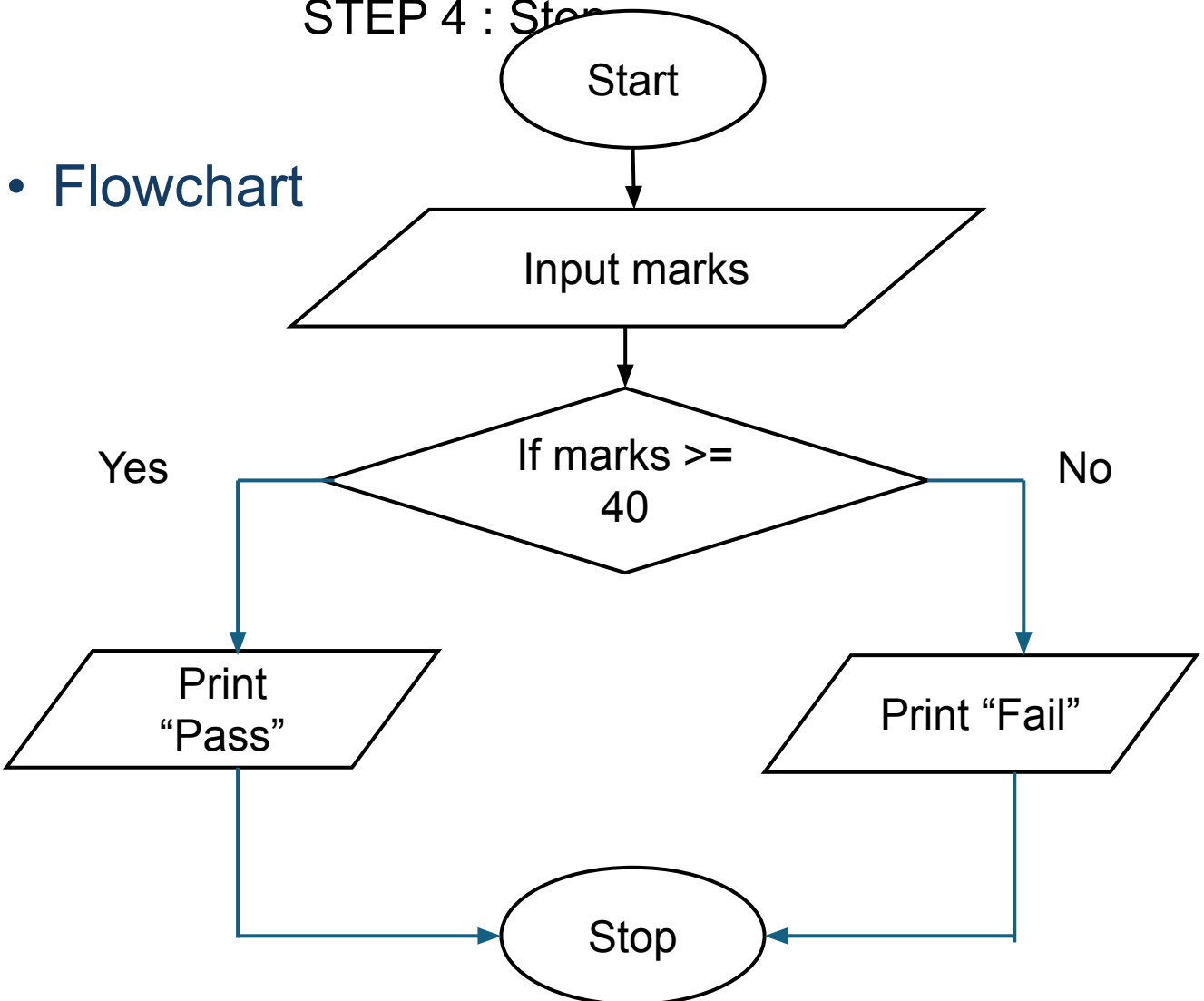
Print "Pass"

Else

Print "Fail"

STEP 4 : Stop

- Flowchart



• Code

```
marks=int(input())
if marks>= 40:
print("Pass")
else:
print("Fail")
```

The screenshot displays the CODETANTRA online IDE interface. The left sidebar contains the problem description for "1.1.5. Student Pass or Fail Status". The main editor shows a Python script that takes an input mark and prints "Pass" or "Fail" based on whether the mark is greater than or equal to 40. The bottom right panel shows the execution results, including a table of test cases and their outcomes.

1.1.5. Student Pass or Fail Status

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

Pass/Fail Criteria:

- A student passes if marks ≥ 40
- A student fails if marks < 40

Input Format:

- Single line contains an integer representing the marks obtained by the student.

Output Format:

- Print "Pass" if the student passed the exam.
- Print "Fail" if the student failed the exam.

Sample Test Cases

Execution Results:

Test Case	Expected Output	Actual Output	Pass
Test case 1	45	45	Pass
Test case 2			
Test case 3			

Summary: 3 out of 3 shown test case(s) passed, 4 out of 4 hidden test case(s) passed.