

### 1.1.1 Area of a circle

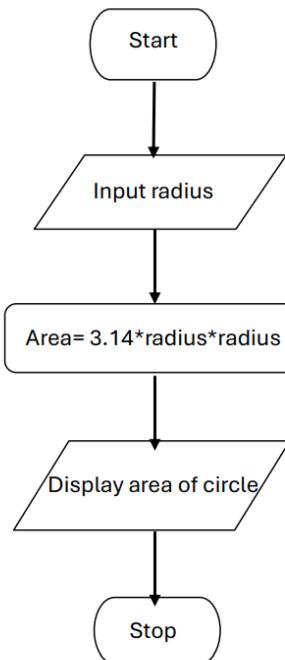
#### ALGORITHM:

- 1.Start
- 2.Read the radius of the circle as a floating-point number.
- 3.Assign the value of  $\pi$  (pi) as 3.14.
- 4.Calculate the area of the circle using the formula:

$$\text{area} = \pi \times \text{radius} \times \text{radius}$$

- 5.Display the calculated area formatted to 4 decimal places.
- 6.Stop

#### FLOWCHART:



#### CODE:

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

## CODETANTRA EXECUTION:

The screenshot shows the CodeTantra IDE interface. On the left, there's a problem statement for "1.1.1. Area of Circle". It asks to write a Python program that calculates the area of a circle given its radius. The input format is a single line of a floating-point number, and the output format is the area formatted to 4 decimal places. A sample test case is provided: radius = 5, expected output is 78.54, and actual output is 78.54.

**Code Editor:**

```
circlearea...
1 radius = float(input())
2 pi = 3.14
3 area = pi * radius * radius
4 print(f"{area:.4f}")
```

**Test Cases:**

- Test case 1: Passed (6 ms)  
Expected output: 3.14  
Actual output: 3.14
- Test case 2: Passed (8 ms)  
Expected output: 35.4493  
Actual output: 35.4493

**Bottom Bar:**

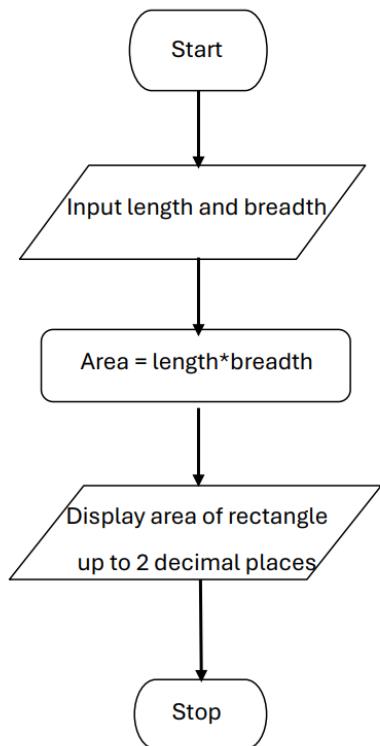
< Prev Reset Submit Next >

## 1.1.2 Area of a rectangle

### ALGORITHM:

- 1 Start
- 2 Read the length of the rectangle as a floating-point number.
- 3 Read the width of the rectangle as a floating-point number.
- 4 Calculate the area using the formula:  
$$\text{area} = \text{length} \times \text{width}$$
- 5
- 6 Display the area of the rectangle formatted to 2 decimal places.
- 7 Stop

### FLOWCHART:



### CODE:

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

## CODETANTRA:

The screenshot shows a challenge titled "1.1.2. Area of Rectangle" on the CodeTantra platform. The challenge instructions state: "Write a Python program to calculate the area of a rectangle given its length and width." Below this, a formula is provided: "Area of Rectangle = Length × Width". The input format specifies that the first line contains the length and the second line contains the width, both as float values. The output format specifies that the area should be printed as a float value formatted to 2 decimal places.

**Code Editor:**

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

**Performance Metrics:**

Average time	Maximum time
0.009 s	0.016 s
8.50 ms	16.00 ms

**Test Cases:**

Test case 1: Expected output 10.5, 5.2, Actual output 10.5, 5.2, Result: Passed (16 ms)

Test case 2: Expected output 54.60, Actual output 54.60, Result: Passed (10 ms)

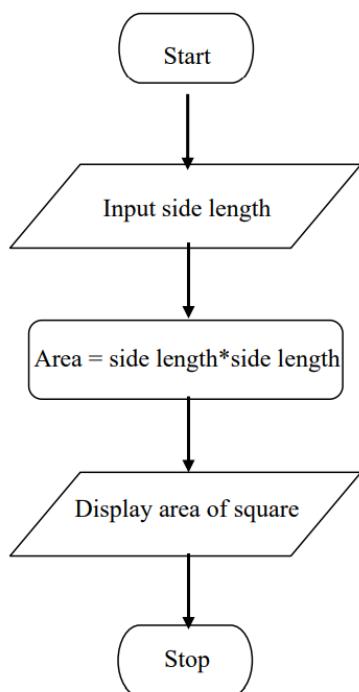
Buttons at the bottom: < Prev, Reset, Submit, Next >

### 1.1.3 Calculate area of a Square

#### ALGORITHM:

- 1 **Start**
- 2 **Read** the side length of the square as an integer.
- 3 **Calculate** the area using the formula:  
$$\text{area} = \text{side} \times \text{side}$$
- 4 **Display** the calculated area.
- 5 **Stop**

#### FLOWCHART:



#### CODE:

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

## CODETANTRA:

The screenshot shows the CodeTantra interface for a challenge titled "1.1.3. Calculate Area of the Square".

**Challenge Details:** 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 =  $\text{side\_length}^2$

**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.

**Code Editor:** The code is named "AreaSqua..." and contains the following Python code:

```
1 side = int(input())
2 area = side * side
3 print(area)
4
```

**Performance Metrics:**

- Average time: 0.007 s
- Maximum time: 0.011 s
- 6.75 ms
- 11.00 ms

**Test Results:**

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

**Test Case 1:** Expected output: 5, Actual output: 25

**Test Case 2:** (Details not visible)

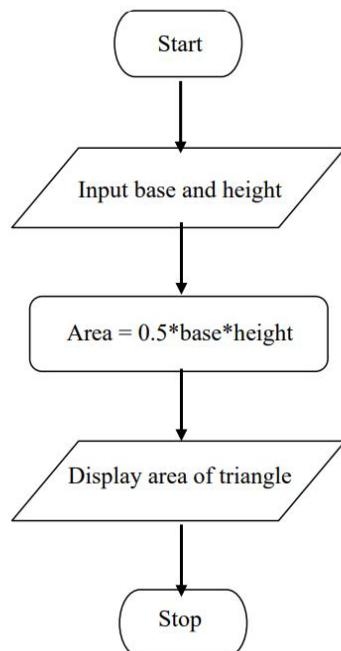
**Buttons:** Terminal, Test cases, < Prev, Reset, Submit, Next >

## 1.1.4Area of Triangle

### ALGORITHM:

1. Start
2. Read the base of the triangle as a floating-point number.
3. Read the height of the triangle as a floating-point number.
4. Calculate the area using the formula:  
$$\text{area} = 0.5 \times \text{base} \times \text{height}$$
5. Display the area formatted to 2 decimal places.
6. Stop

### FLOWCHART:



### CODE:

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

## CODETANTRA:

The screenshot shows a challenge titled "1.1.4. Area of Triangle" on the CodeTantra platform. The challenge description asks the user to write a Python program that prompts the user to enter the triangle's base and height and computes the triangle's area. A formula is provided:  $\text{Area of Triangle} = 0.5 \times \text{base} \times \text{height}$ .

**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.

A sample test case is shown:

```
Sample Test Cases +
```

The code editor contains the following Python code:

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

The execution results show:

Average time: 0.011 s | Maximum time: 0.016 s | 11.00 ms | 16.00 ms

Test case 1 (16 ms):

Expected output	Actual output
6.54	6.54
1.23	1.23
4.02	4.02

Test case 2 (12 ms):

Expected output	Actual output
6.54	6.54
1.23	1.23
4.02	4.02

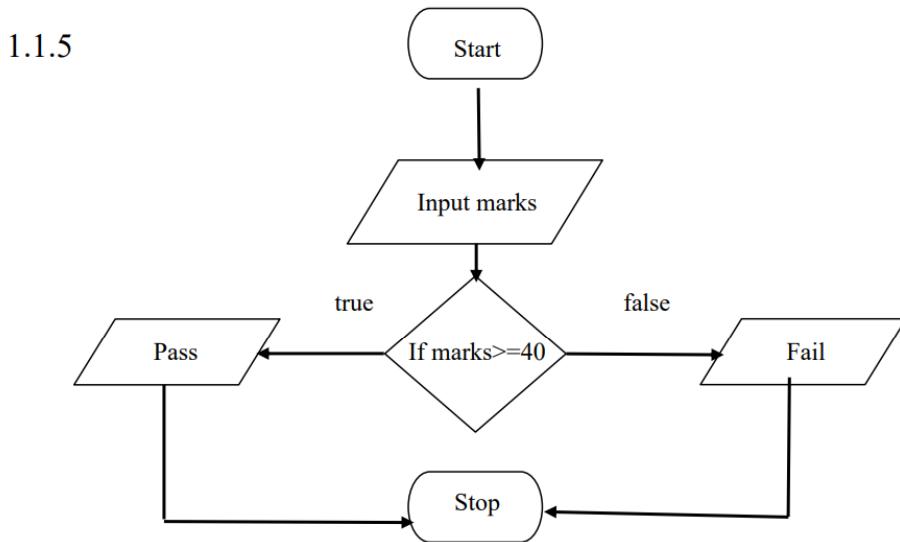
Buttons at the bottom include: < Prev, Reset, Submit, Next >

### 1.1.5. Student Pass or Fail Status

#### ALGORITHM:

1. Start
2. Read the marks obtained by the student as an integer.
3. Check if marks are greater than or equal to 40.
4. If marks  $\geq 40$ , display "Pass".
5. Otherwise, display "Fail".
6. Stop

#### FLOWCHART:



#### CODE:

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

## CODETANTRA:

The screenshot shows a Python code editor on the CodeTantra platform. The code is a simple script named `passOrFail.py` that takes marks as input and prints "Pass" if marks are greater than or equal to 40, otherwise "Fail". The code editor interface includes tabs for `passOrFail.py`, `marks = int(input())`, and `if marks >= 40:`. Below the code are performance metrics: Average time **0.005 s**, Maximum time **0.007 s**, and Execution time **5.14 ms**.

**Test Results:**

- 3 out of 3 shown test case(s) passed**: Expected output `45`, Actual output `45`, Status `Pass`.
- 4 out of 4 hidden test case(s) passed**: Status `Pass`.

At the bottom, there are buttons for `< Prev`, `Reset`, `Submit`, and `Next >`.

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