

EXPERIMENT - 1

1.1.5 Student Pass or Fail

Algorithm

Step 1:- Start

Step 2:- Read the marks obtained by the student.

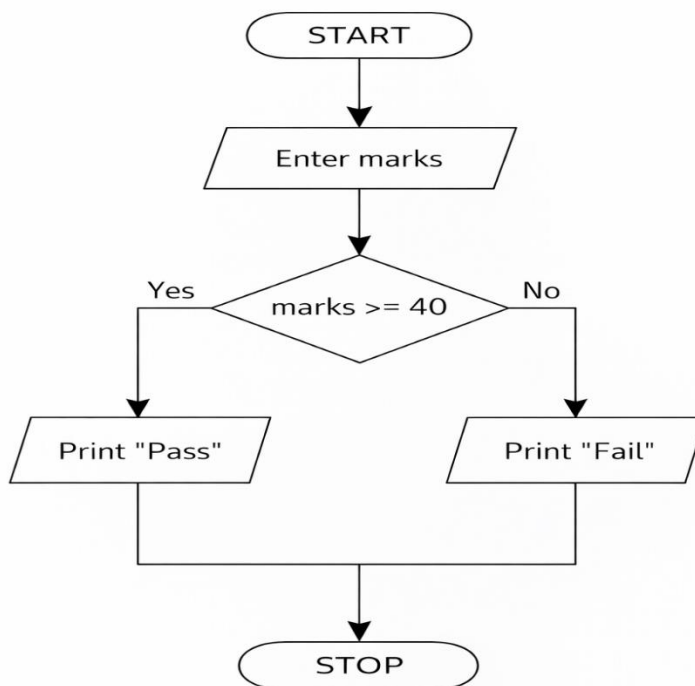
Step 3:- Check whether the marks are greater than or equal to 40.

Step 4:- If marks ≥ 40 , then display "Pass".

Step 5:- Otherwise, display "Fail".

Step 6:- Stop

Flowchart



Python code

EXPERIMENT - 1

```
marks = int(input())
```

```
if marks >= 40:
```

```
    print("Pass")
```

```
else:
```

```
    print("Fail")
```

EXECUTION

The screenshot displays the CODETANTRA online IDE interface. On the left, the problem statement for '1.1.5. Student Pass or Fail Status' is shown, including the criteria for passing (marks ≥ 40) and failing (marks < 40), along with input and output formats. The main editor on the right contains the Python code for the solution. Below the code, the execution results are displayed, showing that 3 out of 3 shown test cases and 4 out of 4 hidden test cases passed. The test cases table shows the expected output '45' and actual output '45' for Test case 1, resulting in 'Pass'.

CODETANTRA Home

harsh.ghaoghare.batch2025@sitnagpur.siu.edu.in Support Logout

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

passOrFa...

```
1 marks=int(input())
2 if marks>=40:
3     print("Pass")
4 else:
5     print("Fail")
```

Average time: 0.009 s, 8.87 ms | Maximum time: 0.030 s, 30.00 ms

3 out of 3 shown test case(s) passed
4 out of 4 hidden test case(s) passed

Test case	Expected output	Actual output	Status
Test case 1	45	45	Pass
Test case 2			
Test case 3			

Terminal Test cases

Prev Reset Submit Next