

EXPERIMENT - 1

Name : Om Kashikar

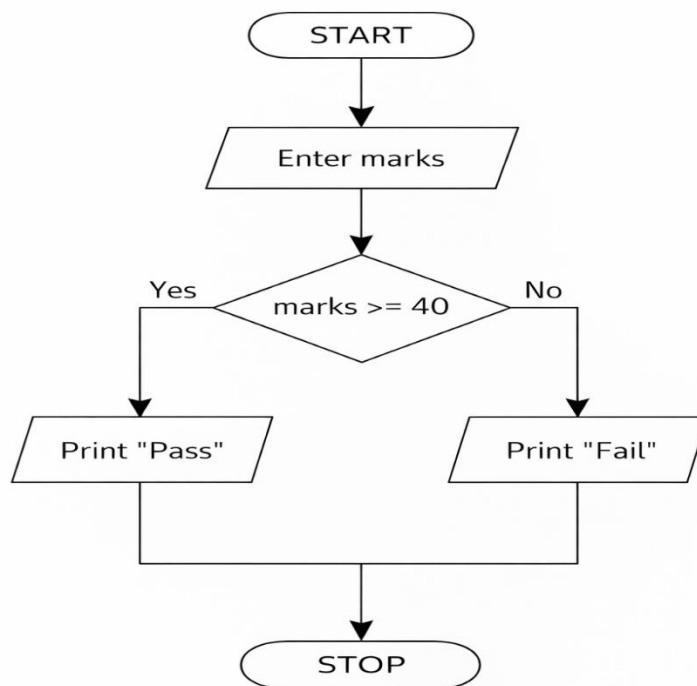
PRN : 25070521170

1.1.5 Student Pass or Fail

Algorithm

1. Start
2. Read the marks obtained by the student.
3. Check whether the marks are greater than or equal to 40.
4. If marks ≥ 40 , then display "Pass".
5. Otherwise, display "Fail".
6. Stop

Flowchart



EXPERIMENT - 1

Python code

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

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

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

```
else:
```

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

EXCECUTION

The screenshot displays the CodeTANTRA 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), and the required input/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. A detailed view of Test case 1 is expanded, showing the input 45 and the output "Pass".

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")
6
```

Average time: 0.003 s, Maximum time: 0.003 s

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

Test case 1 (3 ms)

Expected output	Actual output
45	45
Pass	Pass

Test case 2 (3 ms)

Test case 3 (2 ms)

Terminal | Test cases

< Prev | Reset | Submit | Next >