

EXPERIMENT- 5

5.1.2 Student Grade based on Aggregate

ALGORITHM

Step 1: Start the program.

Step 2: Input marks of four subjects: m1, m2, m3, m4.

Step 3: Calculate the total marks using:

$$\text{total} = m1 + m2 + m3 + m4$$

Step 4: Calculate the aggregate percentage using:

$$\text{aggregate} = (\text{total} / 400) \times 100$$

Step 5: Check the aggregate percentage:

If aggregate > 75, assign grade = Distinction

Step 6: Else if aggregate ≥ 60 and < 75 , assign grade = First Division

Step 7: Else if aggregate ≥ 50 and < 60 , assign grade = Second Division

Step 8: Else if aggregate ≥ 40 and < 50 , assign grade = Third Division

Step 9: Else, assign grade = Fail

Step 10: Display the total marks.

Step 11: Display the aggregate percentage up to two decimal places.

Step 12: Display the grade of the student.

Step 13: End the program.

PYTHON CODE

```
m1, m2, m3, m4 = map(int, input().split())
```

```
total = m1 + m2 + m3 + m4
```

```
aggregate = (total / 400) * 100
```

```
if aggregate > 75:
```

```
    grade = "Distinction"
```

```
elif aggregate >= 60:
```

```
    grade = "First Division"
```

```
elif aggregate >= 50:
```

```
    grade = "Second Division"
```

```
elif aggregate >= 40:
```

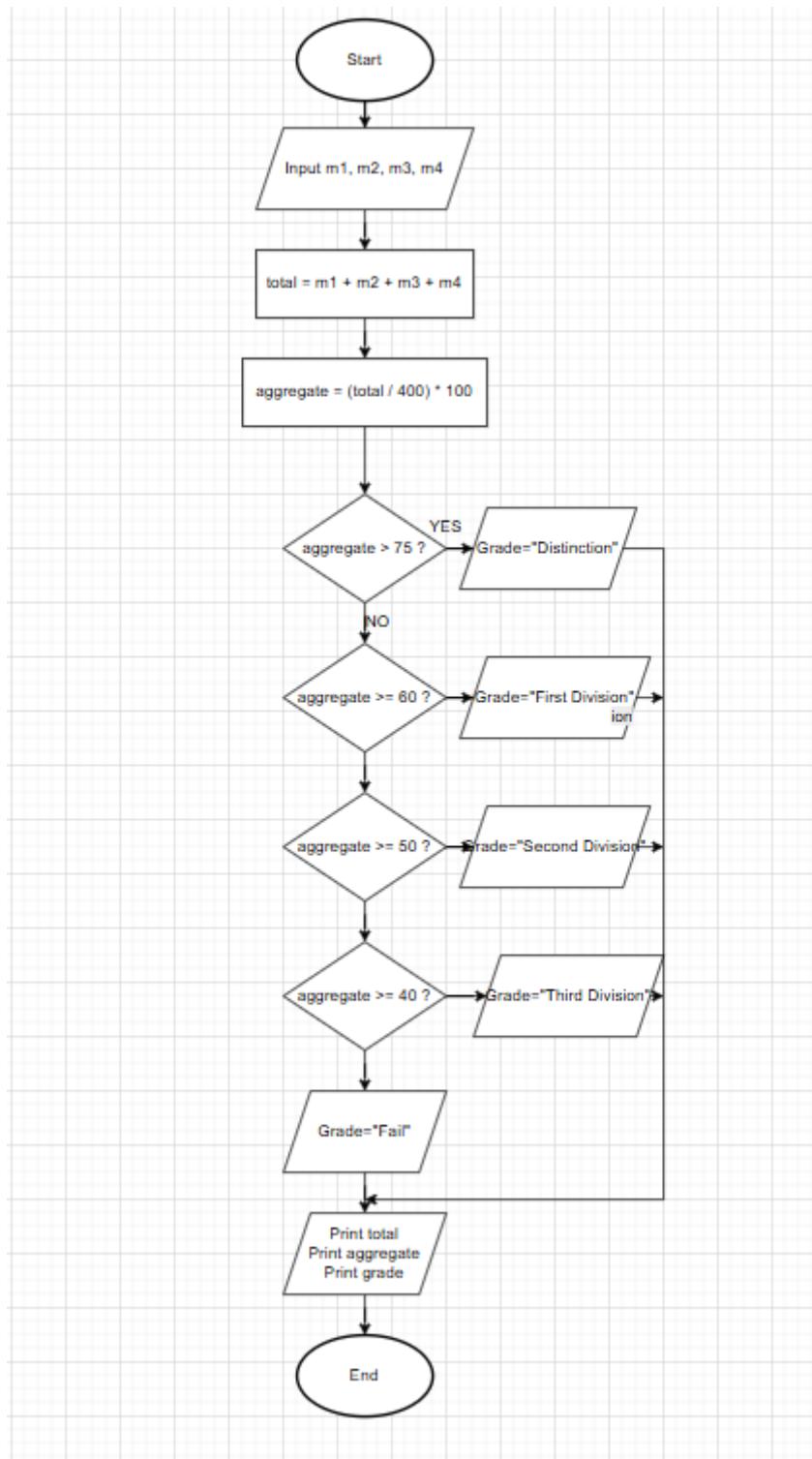
```
    grade = "Third Division"
```

```
else:
```

EXPERIMENT- 5

```
grade = "Fail"  
print(total)  
print(f"{aggregate:.2f}")  
print(grade)
```

FLOWCHART



EXPERIMENT- 5

EXCECUTION

The screenshot shows the CodeTantra IDE interface. The top bar displays the user's name, email, support link, and a logout button. The main area has tabs for 'studentG...' and 'Explore'. The code editor contains a Python script named 'studentG...' which calculates student grades based on aggregate marks. The script uses a list comprehension to convert input strings into integers, calculates the total marks, and then determines the grade based on the aggregate percentage using nested if-elif statements. The output section shows that all 10 test cases (5 sample and 5 hidden) passed. The execution time was 0.003 seconds. Below the output, there are sections for 'Test case 1' and 'Test case 2' showing expected and actual outputs, and a terminal window.

```
marks= list(map(int, input().split()))
total = sum(marks)
aggregate= total / 4
if aggregate > 75:
    grade = "Distinction"
elif aggregate >= 60:
    grade = "First Division"
elif aggregate >= 50:
    grade = "Second Division"
elif aggregate >= 40:
    grade = "Third Division"
else:
    grade = "Fail"
```

Average time: 0.003 s Maximum time: 0.009 s
3.10 ms 9.00 ms 5 out of 5 shown test case(s) passed
5 out of 5 hidden test case(s) passed

Test case 1 (ms)
Expected output: 85 99 78 88
Actual output: 85 99 78 88
341 341
85.25 85.25
Distinction Distinction

Test case 2 (ms)
Terminal Test cases