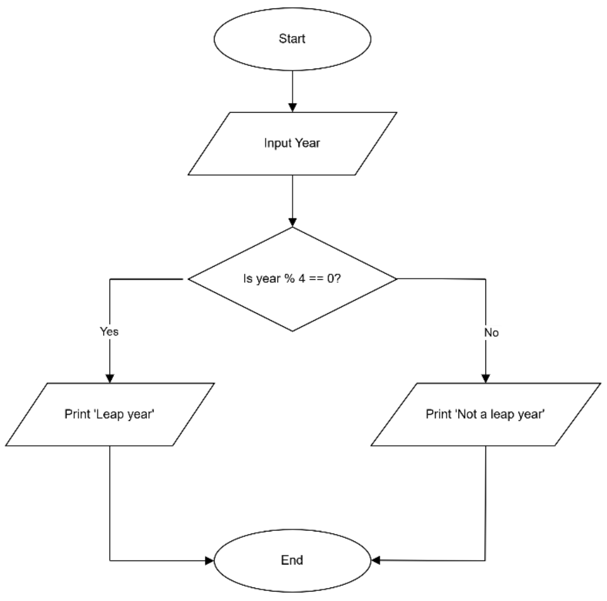
5.1.1. Leap Year Checker

Algorithm:

```
1. START
2. INPUT year
3. IF year % 4 == 0 THEN
      PRINT "Leap year"
   ELSE
      PRINT "Not a leap year"
4. END
```

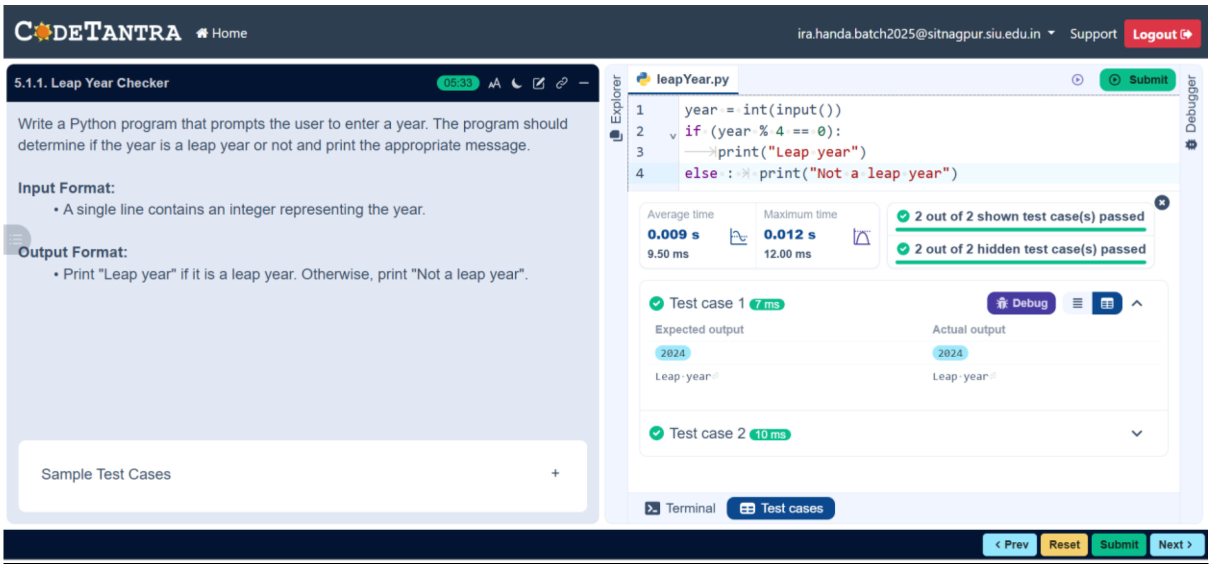
Flowchart:



Code:

```
year = int(input())
if year % 4 == 0:
   print("Leap year")
else:
   print("Not a leap year")
```

Code Tantra Execution:



5.1.1. Leap Year Checker

Write a Python program that prompts the user to enter a year. The program should determine if the year is a leap year or not and print the appropriate message.

Input Format:

- A single line contains an integer representing the year.

Output Format:

- Print "Leap year" if it is a leap year. Otherwise, print "Not a leap year".

Sample Test Cases +

leapYear.py

```python
year = int(input())
if (year % 4 == 0):
    print("Leap year")
else :  print("Not a leap year")
```

Average time 0.009 s 9.50 ms

Maximum time 0.012 s 12.00 ms

2 out of 2 shown test case(s) passed

2 out of 2 hidden test case(s) passed

Test case 1 7 ms

Expected output	Actual output
2024	2024
Leap year	Leap year

Test case 2 10 ms

5.1.2. Student Grade Based on Aggregate

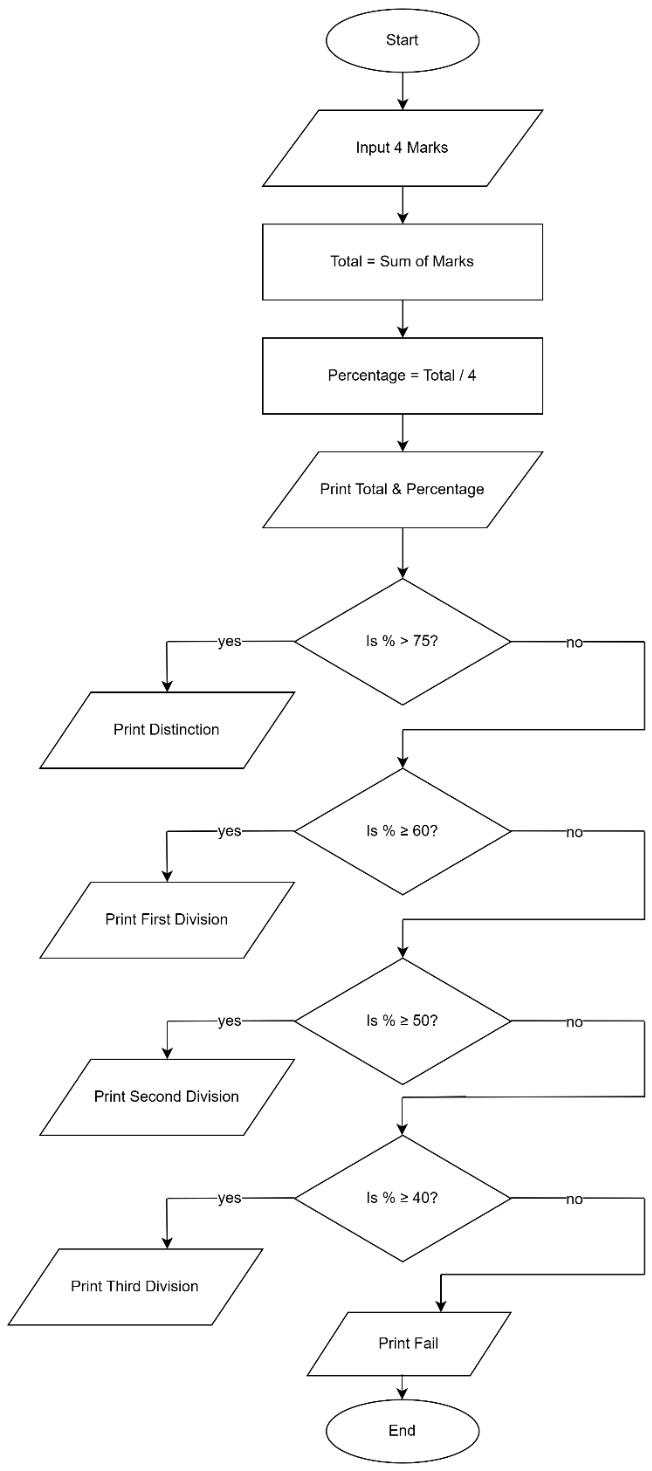
Algorithm:

1. **START**
2. **INPUT** four marks (a, b, c, d) from user
3. **CALCULATE** total_marks = a + b + c + d
4. **PRINT** total_marks
5. **CALCULATE** aggregate = total_marks / 4
6. **PRINT** aggregate with 2 decimal places
7. **CHECK CONDITIONS:**
8. IF aggregate > 75: PRINT "Distinction"
9. ELSE IF aggregate >= 60: PRINT "First Division"
10. ELSE IF aggregate >= 50: PRINT "Second Division"
11. ELSE IF aggregate >= 40: PRINT "Third Division"
12. ELSE: PRINT "Fail"
13. **END**

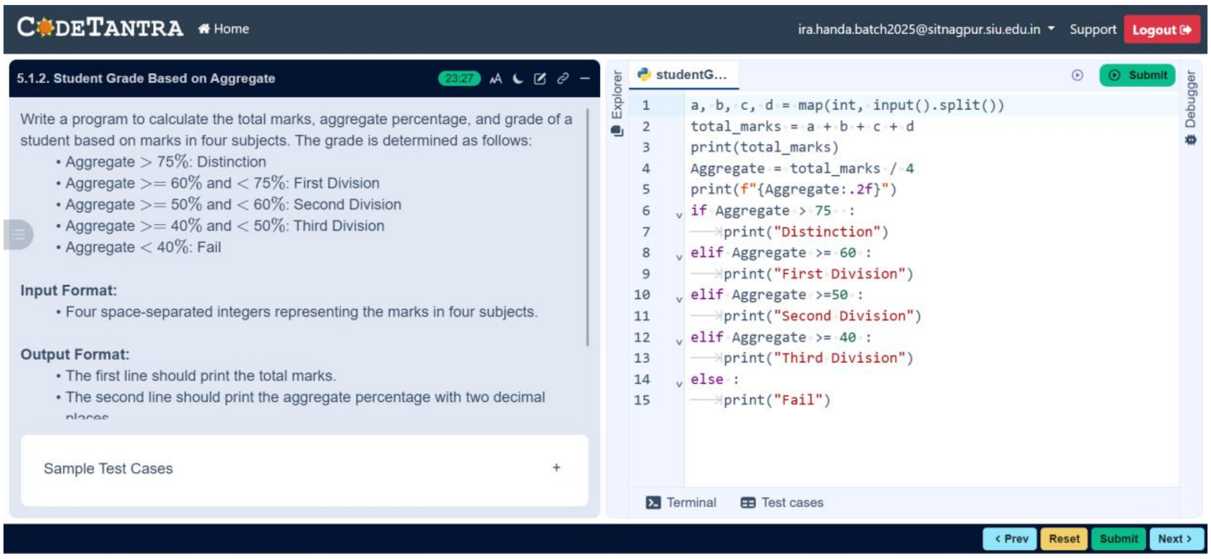
Code:

```
a, b, c, d = map(int, input().split())

total_marks = a + b + c + d

print(total_marks)

aggregate = total_marks / 4

print(f"{aggregate:.2f}")

if aggregate > 75:

  print("Distinction")

elif aggregate >= 60:

  print("First Division")

elif aggregate >= 50:

  print("Second Division")

elif aggregate >= 40:
  print("Third Division")
else:
  print("Fail")
```

Flowchart:



Code Tantra Execution:



5.1.2. Student Grade Based on Aggregate

Write a program to calculate the total marks, aggregate percentage, and grade of a student based on marks in four subjects. The grade is determined as follows:

- Aggregate $> 75\%$: Distinction
- Aggregate $>= 60\%$ and $< 75\%$: First Division
- Aggregate $>= 50\%$ and $< 60\%$: Second Division
- Aggregate $>= 40\%$ and $< 50\%$: Third Division
- Aggregate $< 40\%$: Fail

Input Format:

- Four space-separated integers representing the marks in four subjects.

Output Format:

- The first line should print the total marks.
- The second line should print the aggregate percentage with two decimal

Sample Test Cases +

```python
a, b, c, d = map(int, input().split())
total_marks = a + b + c + d
print(total_marks)
Aggregate = total_marks / 4
print(f"{Aggregate:.2f}")
if Aggregate > 75  :
    print("Distinction")
elif Aggregate >= 60 :
    print("First Division")
elif Aggregate >=50 :
    print("Second Division")
elif Aggregate >= 40 :
    print("Third Division")
else :
    print("Fail")
```