

# Flowcharts

## Steps to solve any coding question:

1. Analyze the problem
2. Think of solution
3. Pen down the solution
4. Code the solution

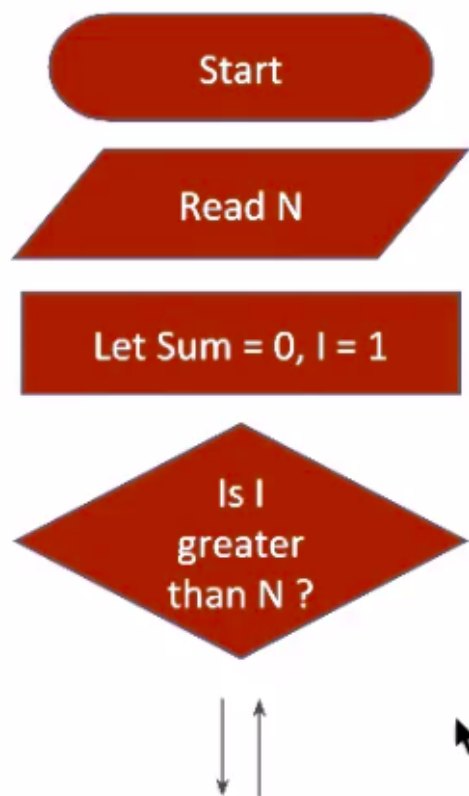
## Flowchart:

- Diagrammatic representation of solution to a given problem
- It break down the solution into small pieces and and display them visually

## Advantages:

- debugging is easy
- language independent
- reduce complexity of problem to reduce into smaller pieces

## Components of Flowchart:



**Terminator**

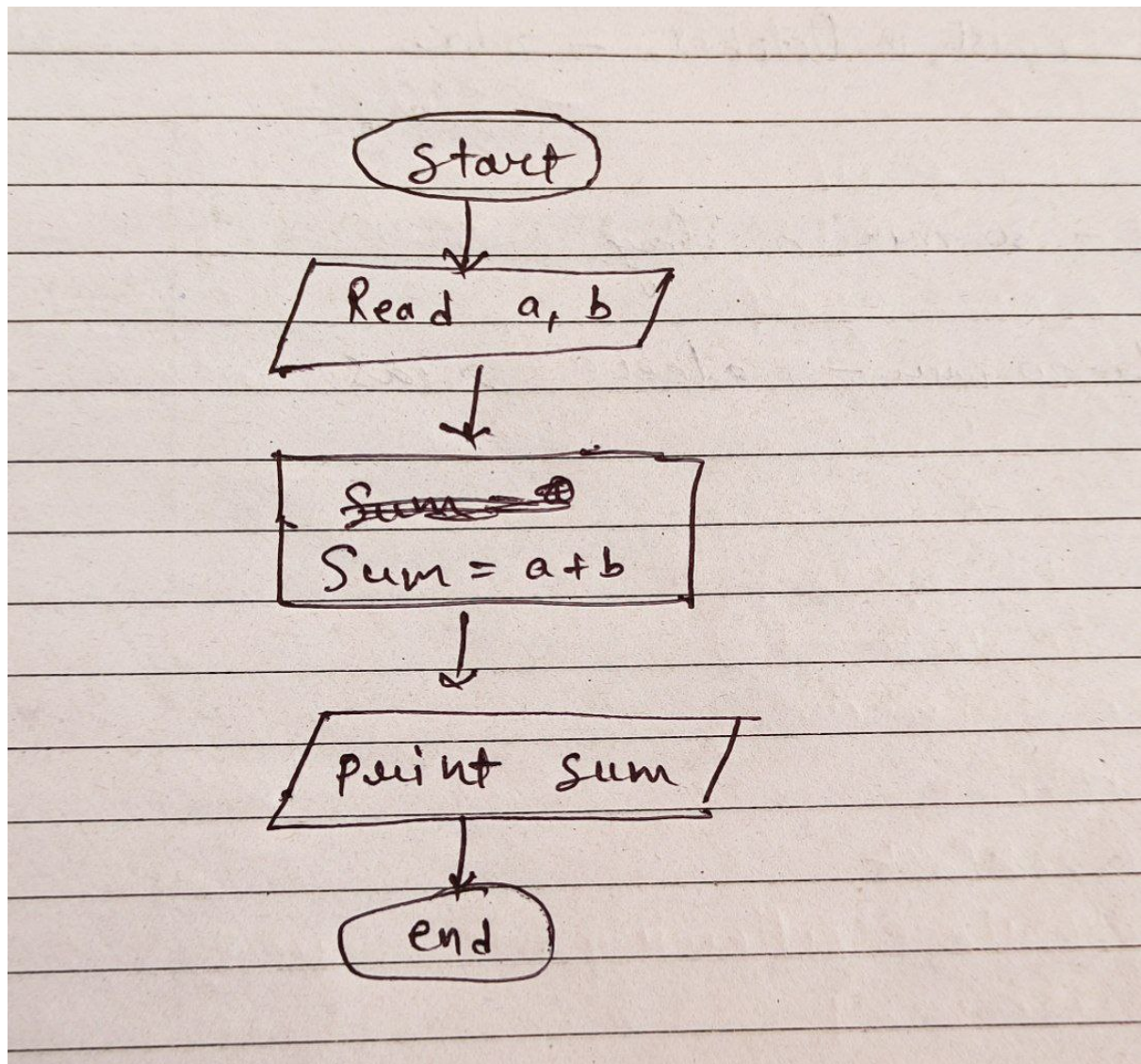
**Input / Output**

**Process**

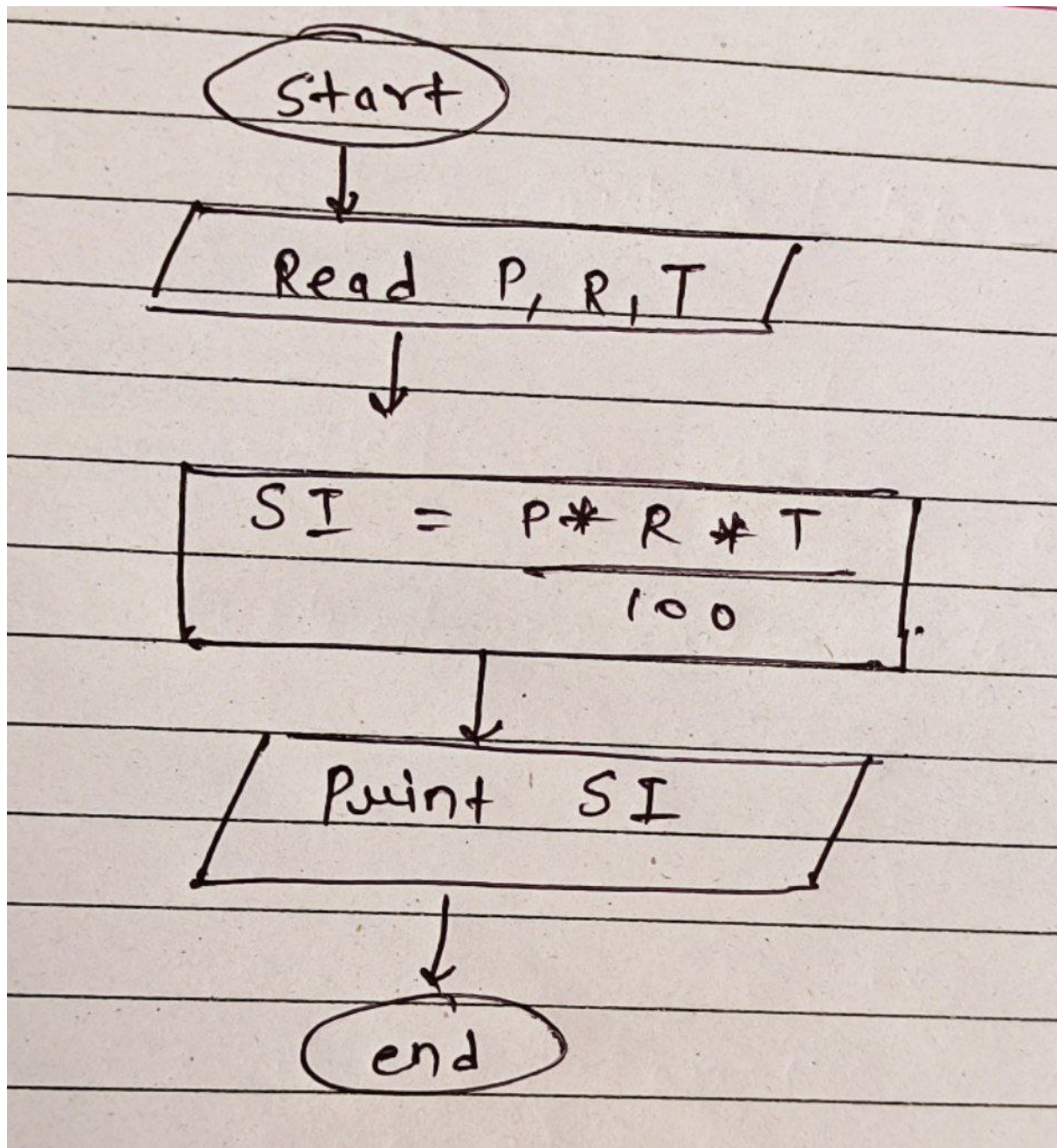
**Decision**

**Arrow**

**Ques: Adding Two numbers**



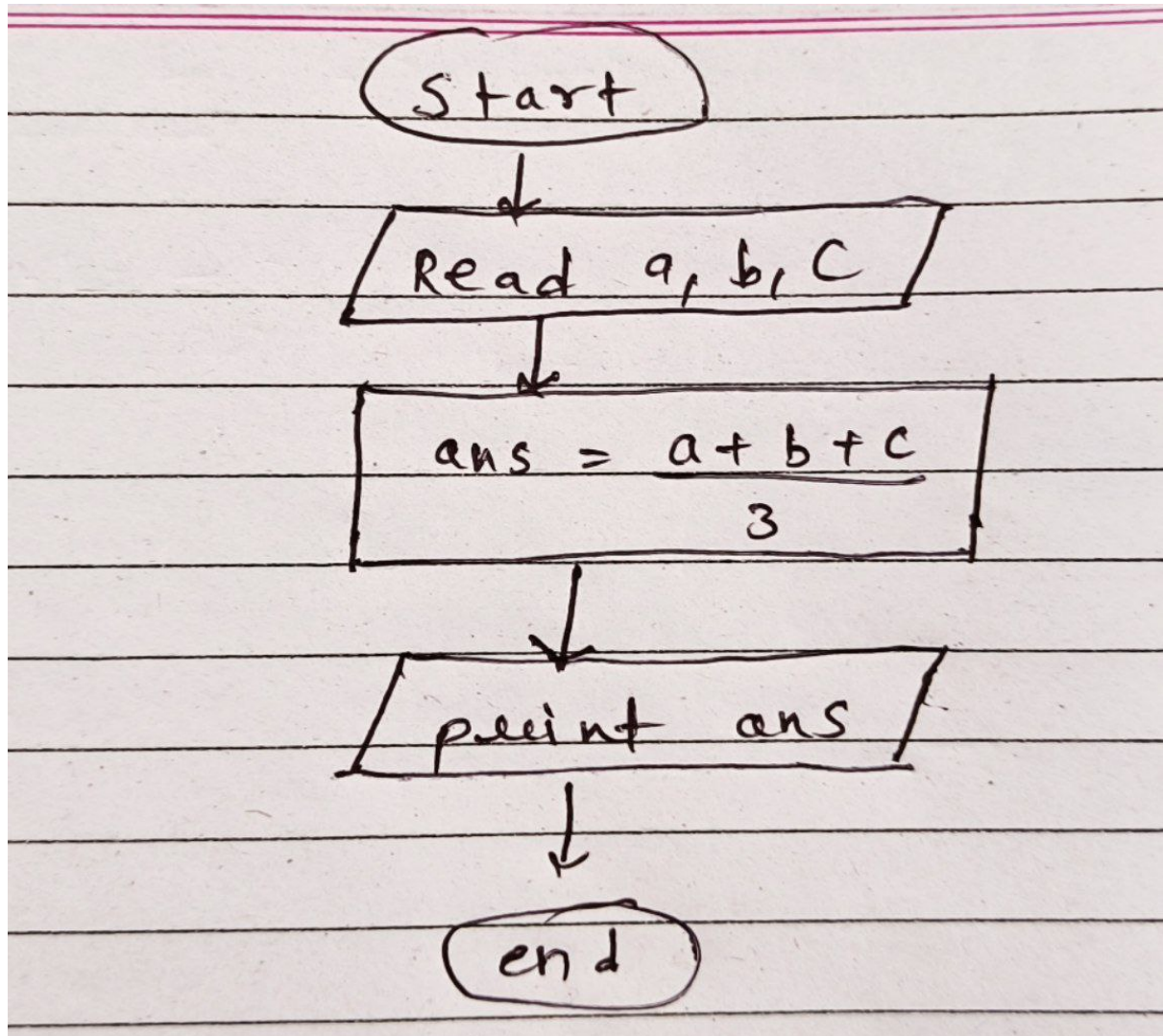
**Ques: Find simple interest**



**Ques: Average of three numbers**

You are given three numbers. You need to calculate and print their average value. Draw a flowchart for this process.

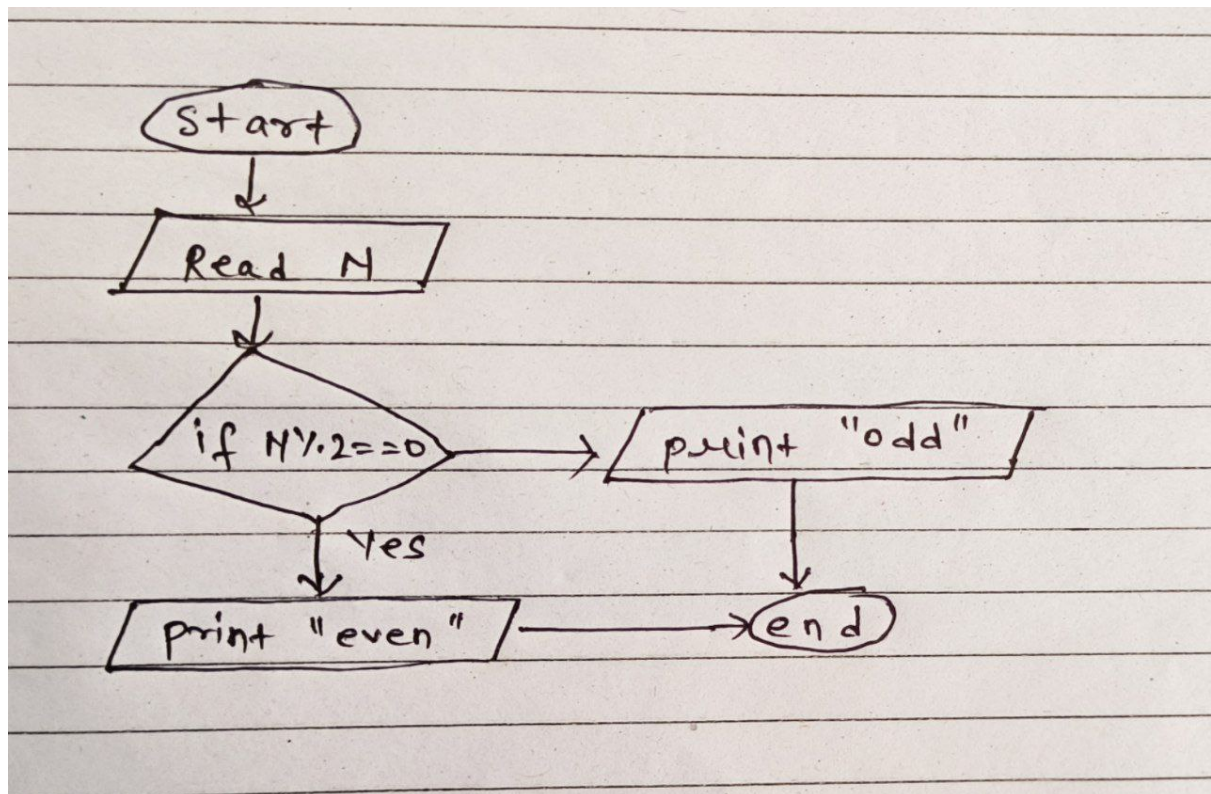




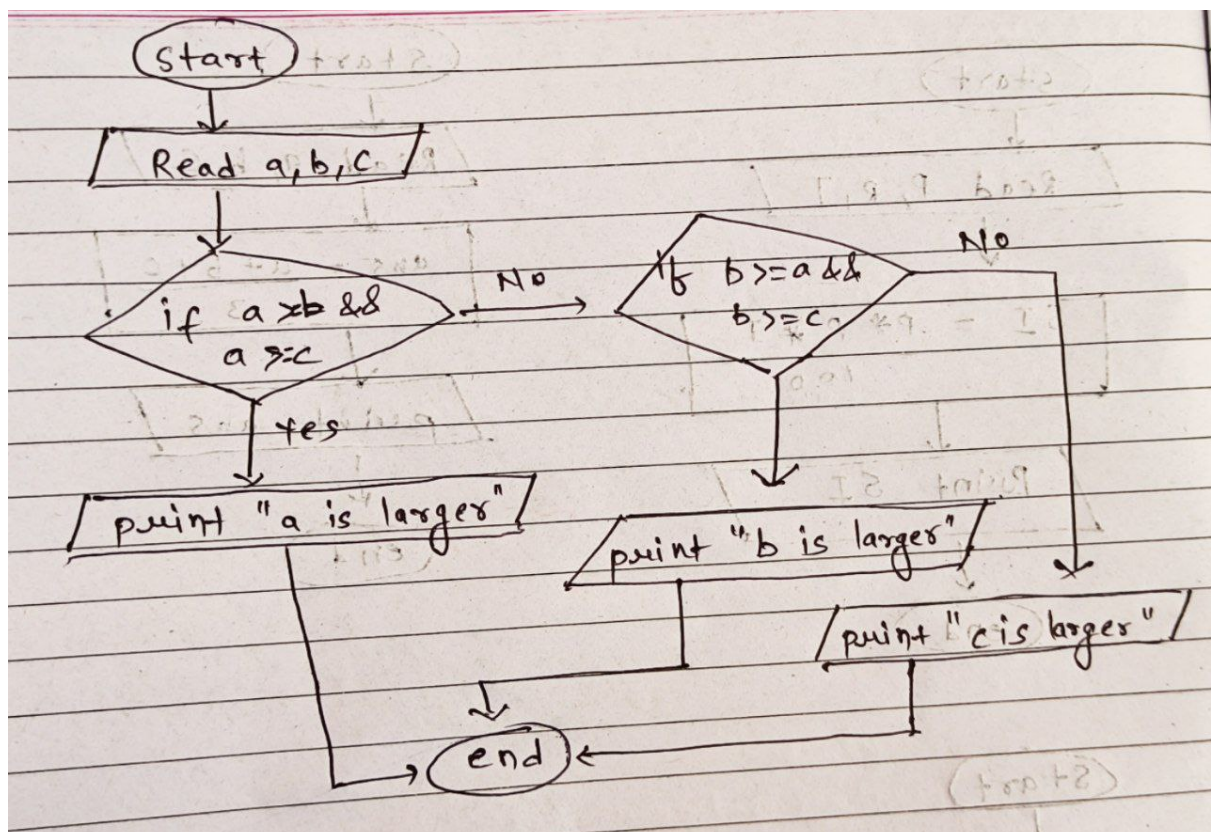
### Decision Making Algorithm

when we have some condition then in flowchart we represent it by rhombus diagram.

**Ques; Given number is odd or even**



Ques: Largest of three numbers





### Ques: Check Number

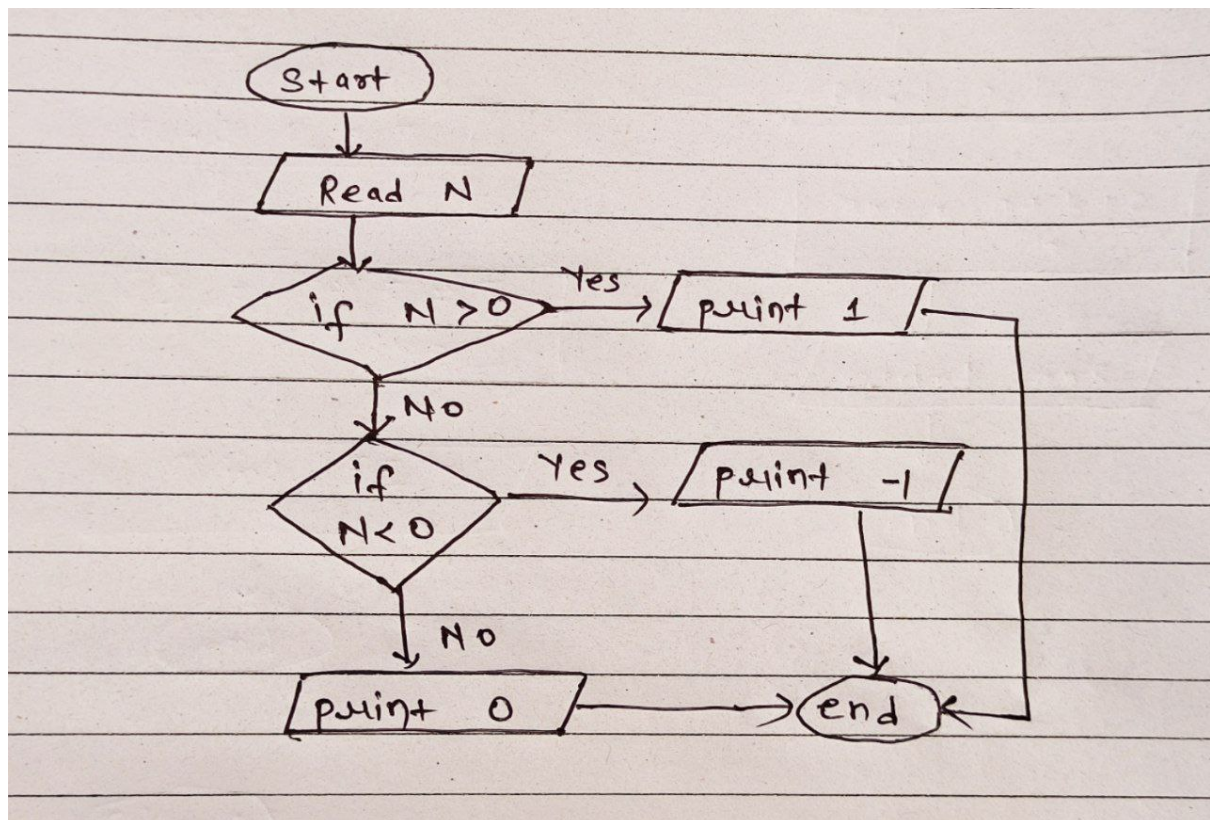
You are given a single number. You need to print one of the following outputs according to the number's nature.

Print 1, if the number is positive

Print -1, if it's negative

Print 0, if it's equal to 0

Draw a flowchart for this process.

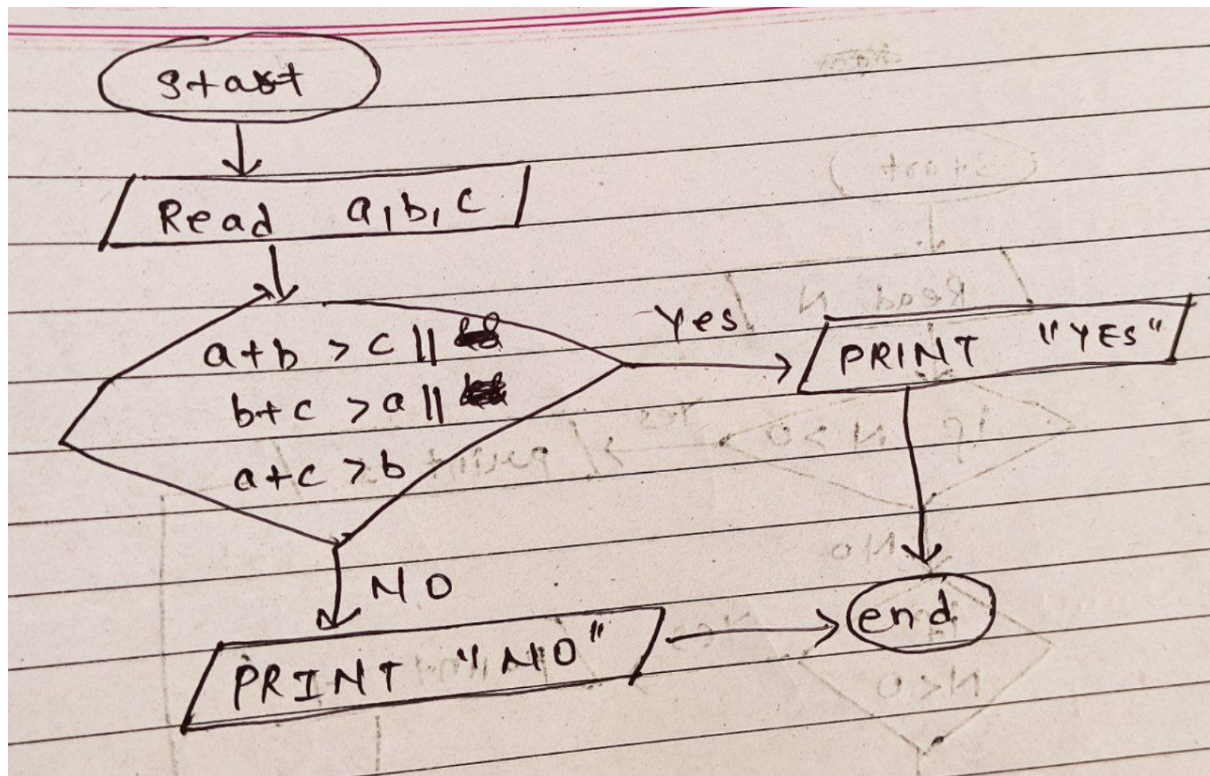


### Ques: Valid Triangle

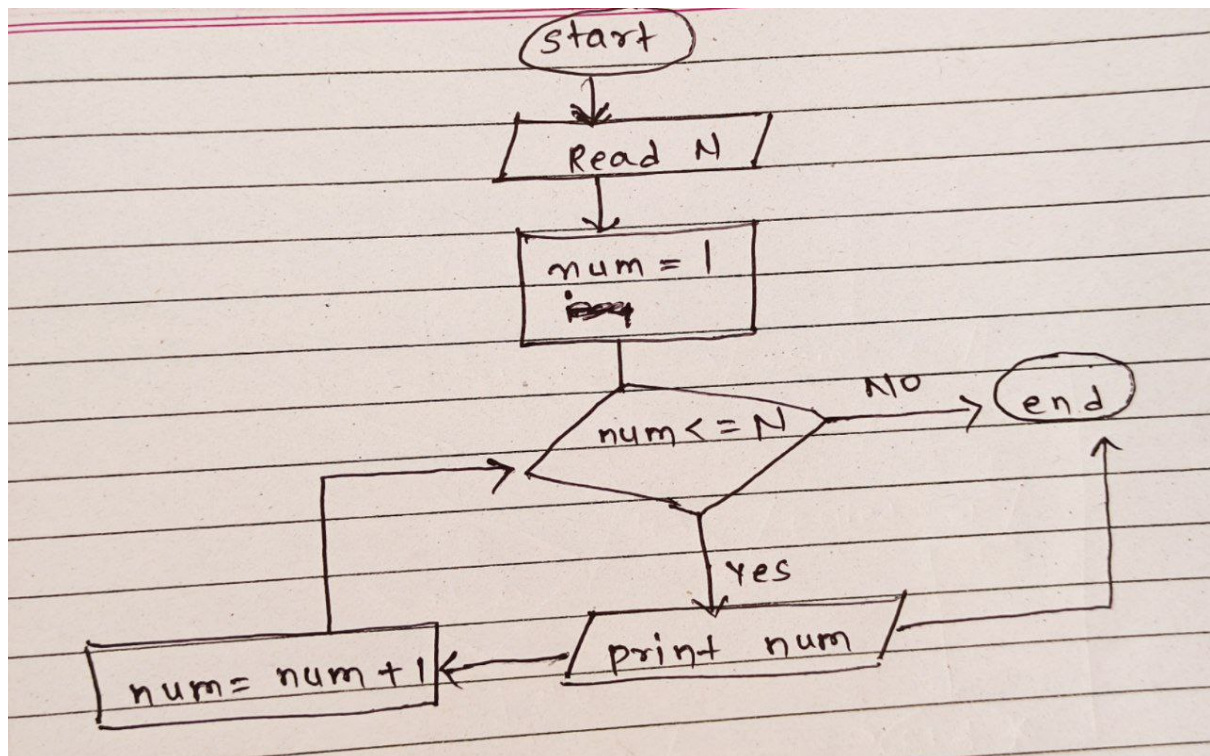
You are given 3 numbers. Each number represents the length of a line. You need to figure out whether these lines can form a valid triangle.

If a valid triangle can be formed, print "Yes", otherwise print "No".

Draw a flowchart for this process.



Ques: Print 1 to N



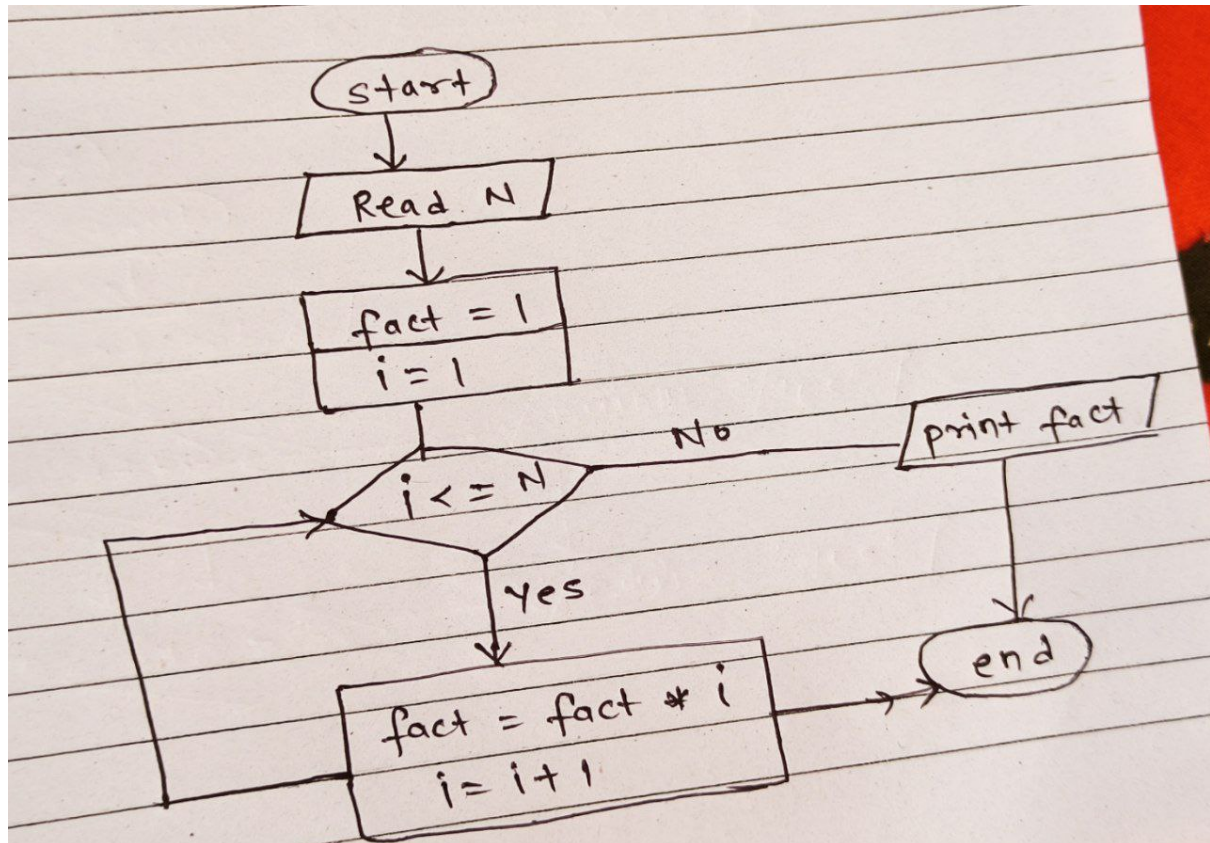


**Ques: Find Product**

You are given a single non-negative integer,  $N$ . You need to calculate and print  $N$  factorial ( $N!$ )

$N$  factorial is defined as the product of all integers from 1 to  $N$  (both inclusive)

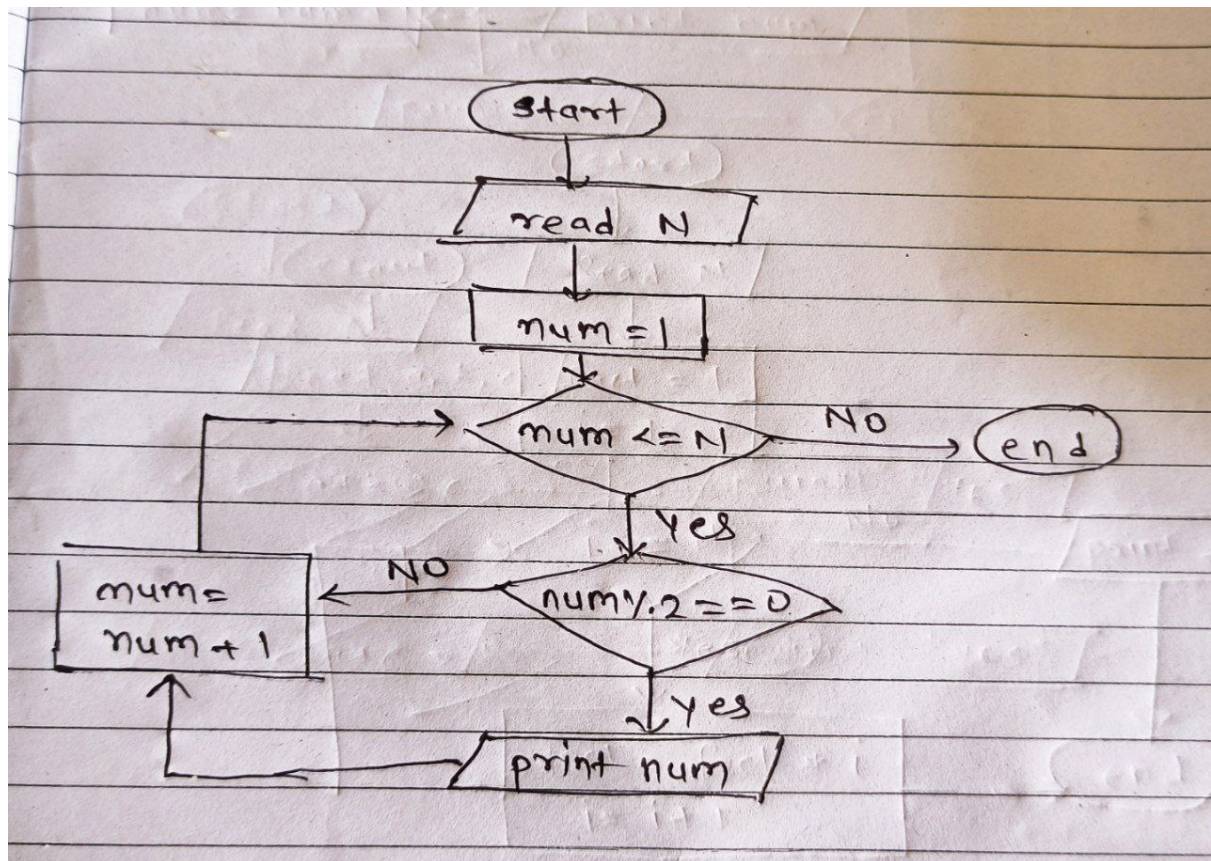
Draw a flowchart for this process



**Ques: Print Even Numbers**

You are given a single positive integer,  $N$ . You need to print all even integers that occur between 1 and  $N$  (both inclusive).

Draw a flowchart for this process



**Ques:** Check number is prime or not

**Note:**



prime number:

logic :

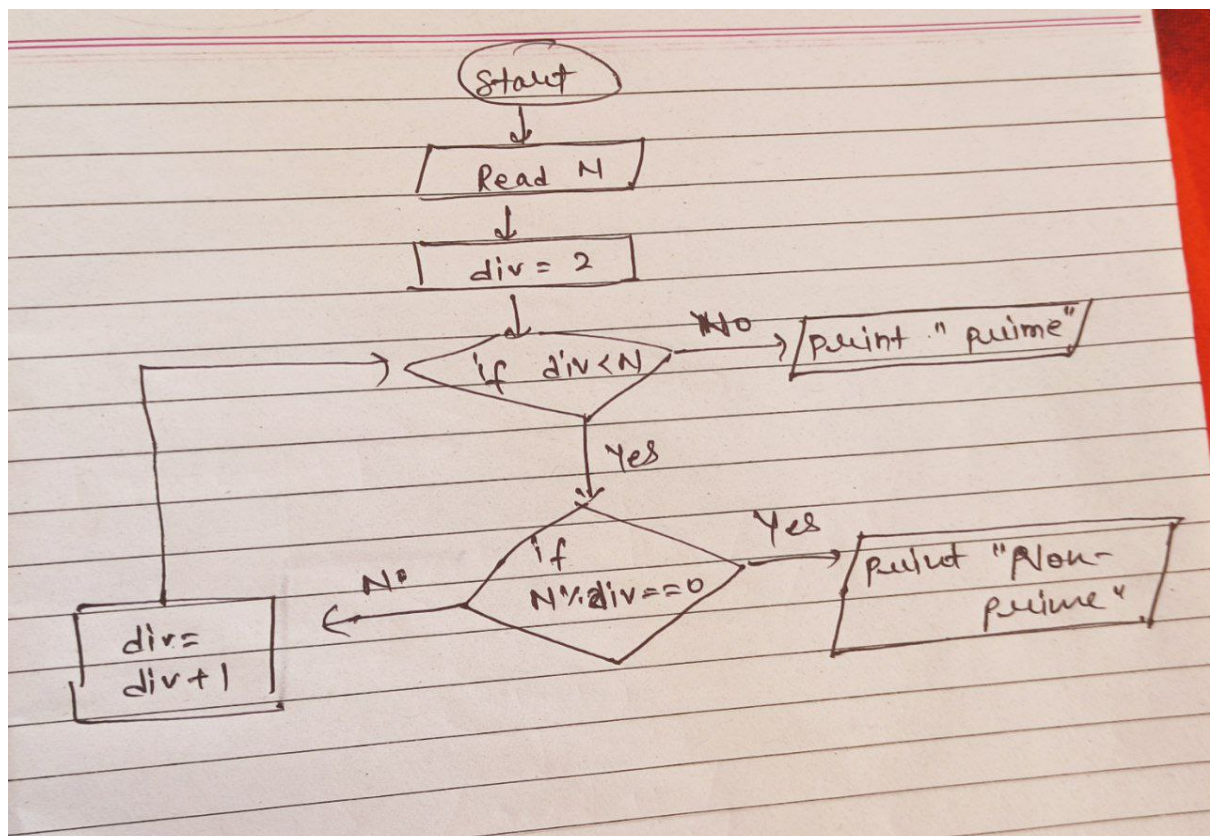
i) Prime no. is divisible by 1 and itself only.

Approach 1 :- ii) check from 2 to  $n-1$ . if  $n$  is not divisible ~~for~~ in these range then " $n$ " is prime no.

Approach 2 :- Optimised Approach  
check from 2 to  $n/2$   
condition to check ( $n \div n$ ).

```
for (i=2; i < n/2; i++) {  
    }  
}
```

flowchart:





Ques: Largest of N numbers

