DSA

Day-1

machine/Binary language
↓
0's and 1's

Binary number → Base 2
Decimal no → Base 10

9523
↓
$9\times1000+5\times100+2\times10+3\times1$
$9\times10^3+5\times10^2+2\times10^1+3\times10^0$

$13 \Rightarrow 1\times10^1+3\times10^0$ ⟷ Binary to decimal
1101
↓
$1\times2^3+1\times2^2+0\times2^1+1\times2^0 \Rightarrow 13$

Compilers/Interpreters → tools which convert source code into machine understandable code.
written by programmers
binary code

1) Assembly lang (1947)
2) FORTRAN
3) COBALT
4) BASIC
5) C
   └ Unix based system
6) C++
7) Java

Eg: Decimal to binary

$(54)_{10} \rightarrow (0b110110)_2$

2|54
2|27-0
2|13-1
2|6-1
2|3-0
2|1-1

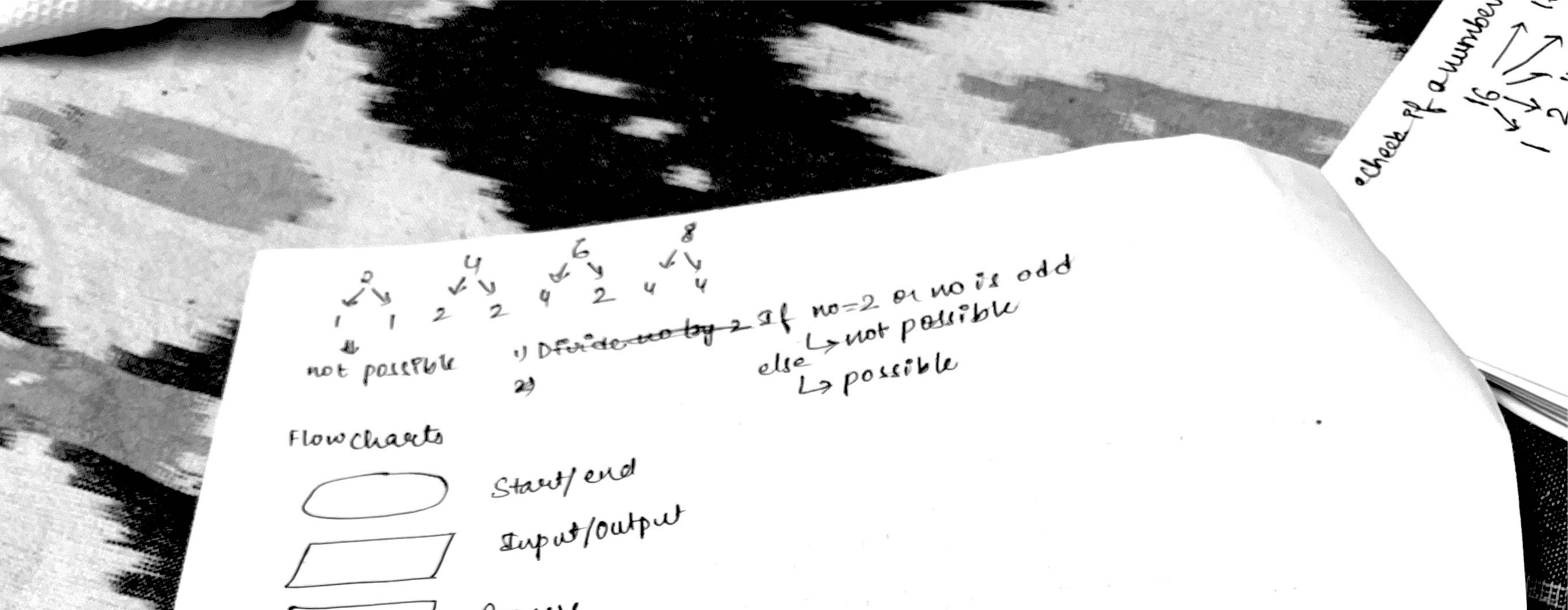
- Given a number, check whether no is even or odd?
  1) Divide the number by 2
  2) If the remainder is 0 then number is even else number is odd
- You have a watermelon weights wkg. Cut the watermelon to exactly 2 pieces & both parts should weigh even no of kilos

  Eg: 12kg → Possible
  ↳ 6kg 6kg
  8kg 4kg

  10kg → possible
  ↳ 6kg 4kg
  ↳ 8kg 2kg

~~if~~

1) Divide w by 2
2) If the remainder is 1
   ↳ not possible
   else ↳ possible



check if a number is power of 2

16 → 8 → 4 → 2 → 1

2 → 1

4 → 2 → 1

6 → 3

8 → 4 → 2 → 1

1) Divide no by 2 if no = 2 or no is odd
   else ↳ not possible
   ↳ possible

2)

not possible

Flowcharts

Start/end

Input/output

Process

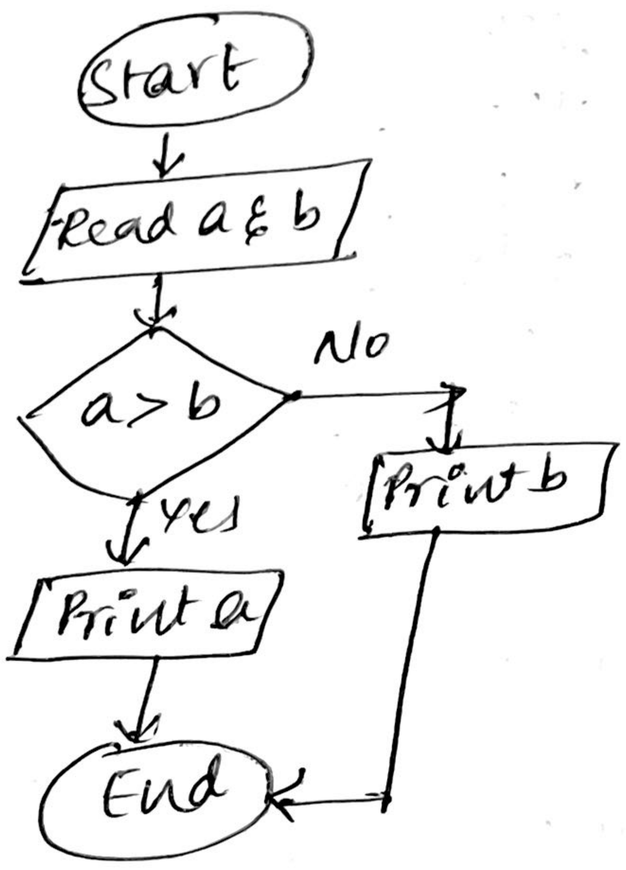
Decision

Arrow

- Get 2 no's and print the max

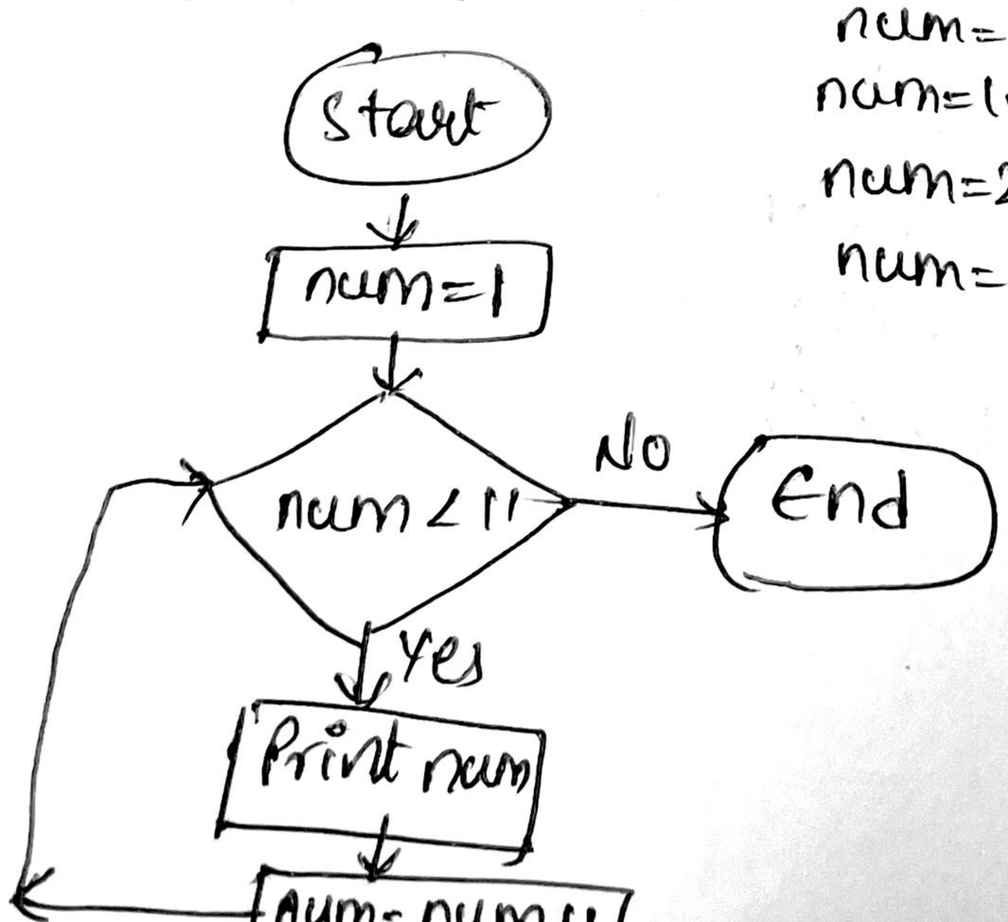
  eg: 7, (12) 8, (15)

1) Read a and b
2) If a > b
   ↳ print a
   else ↳ print b

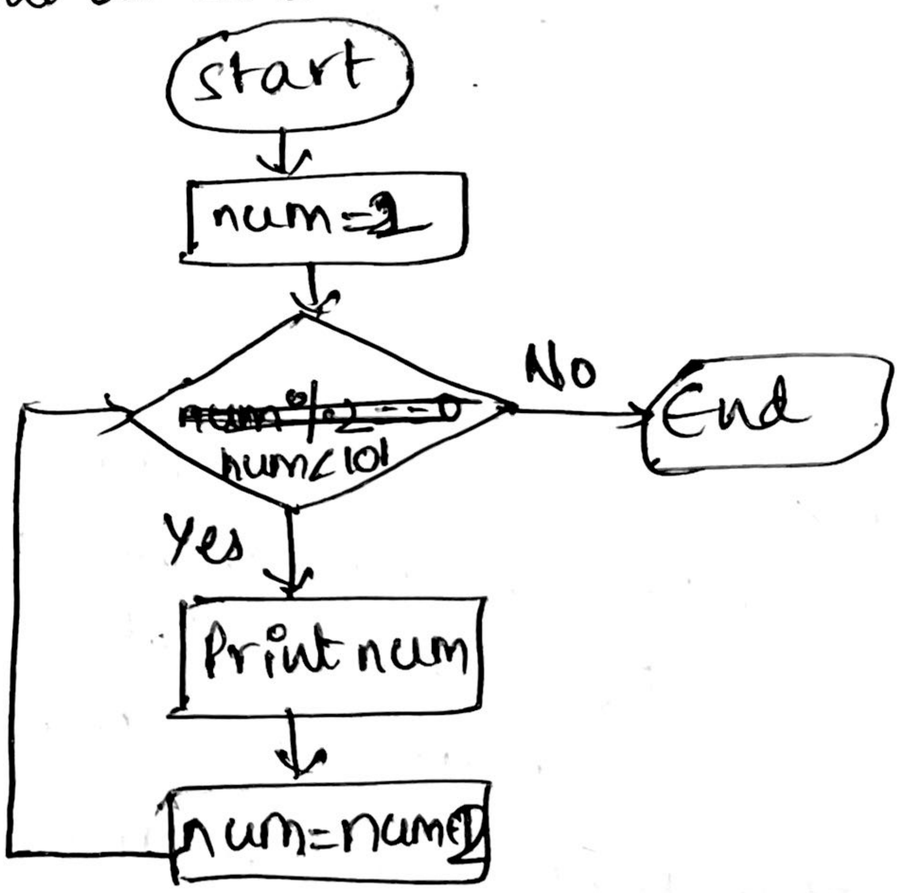


- Print no's from 1 to 10

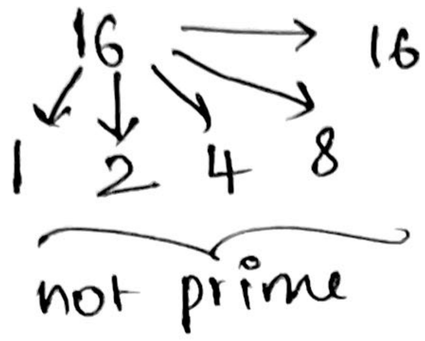
num = 1
num = 1 + 1 = 2
num = 2 + 1 = 3
num = 3 + 1 = 4

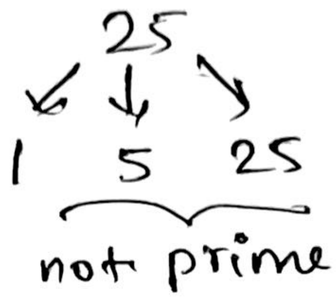


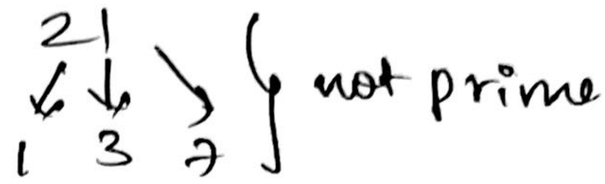
- Print all even no's from 1 to 100

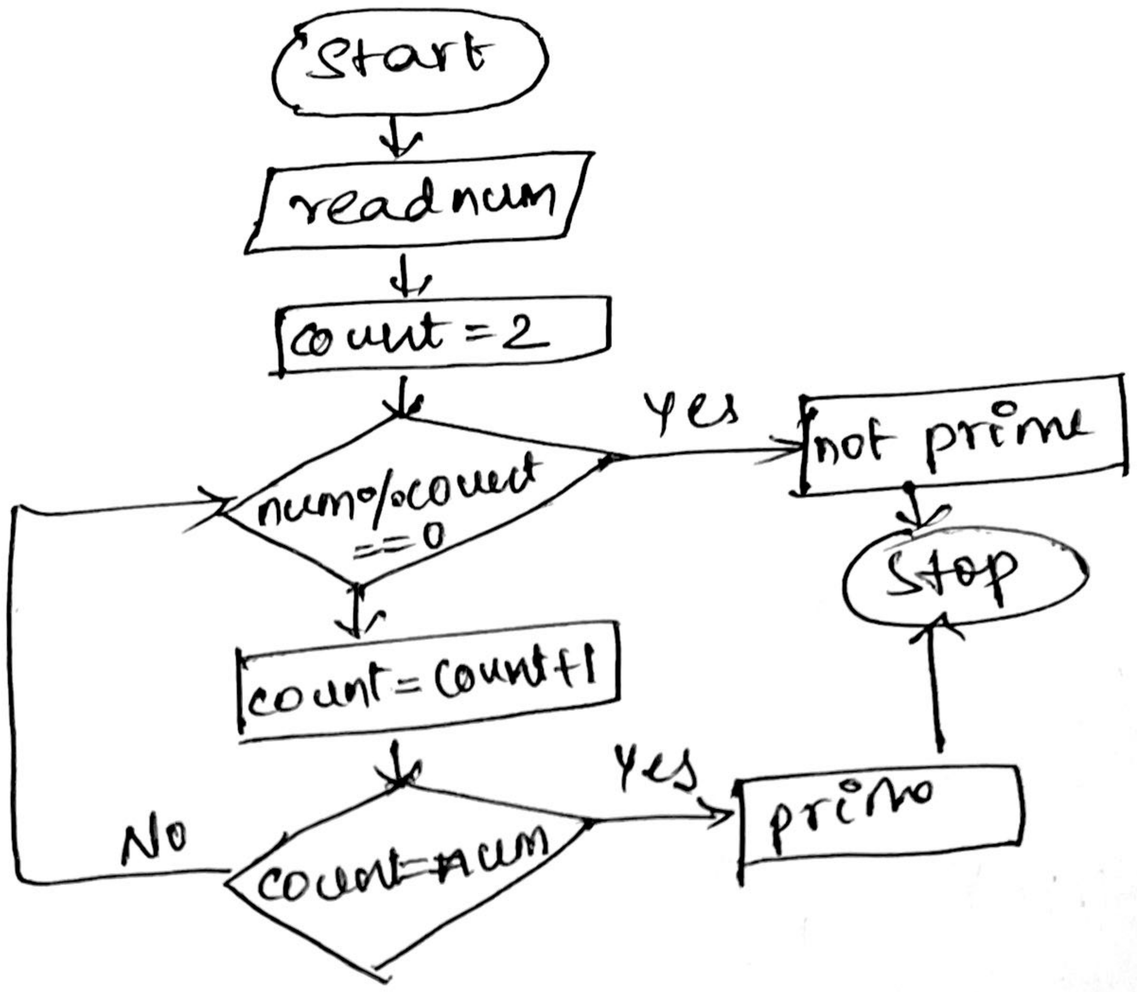


• check if a number is a prime or not



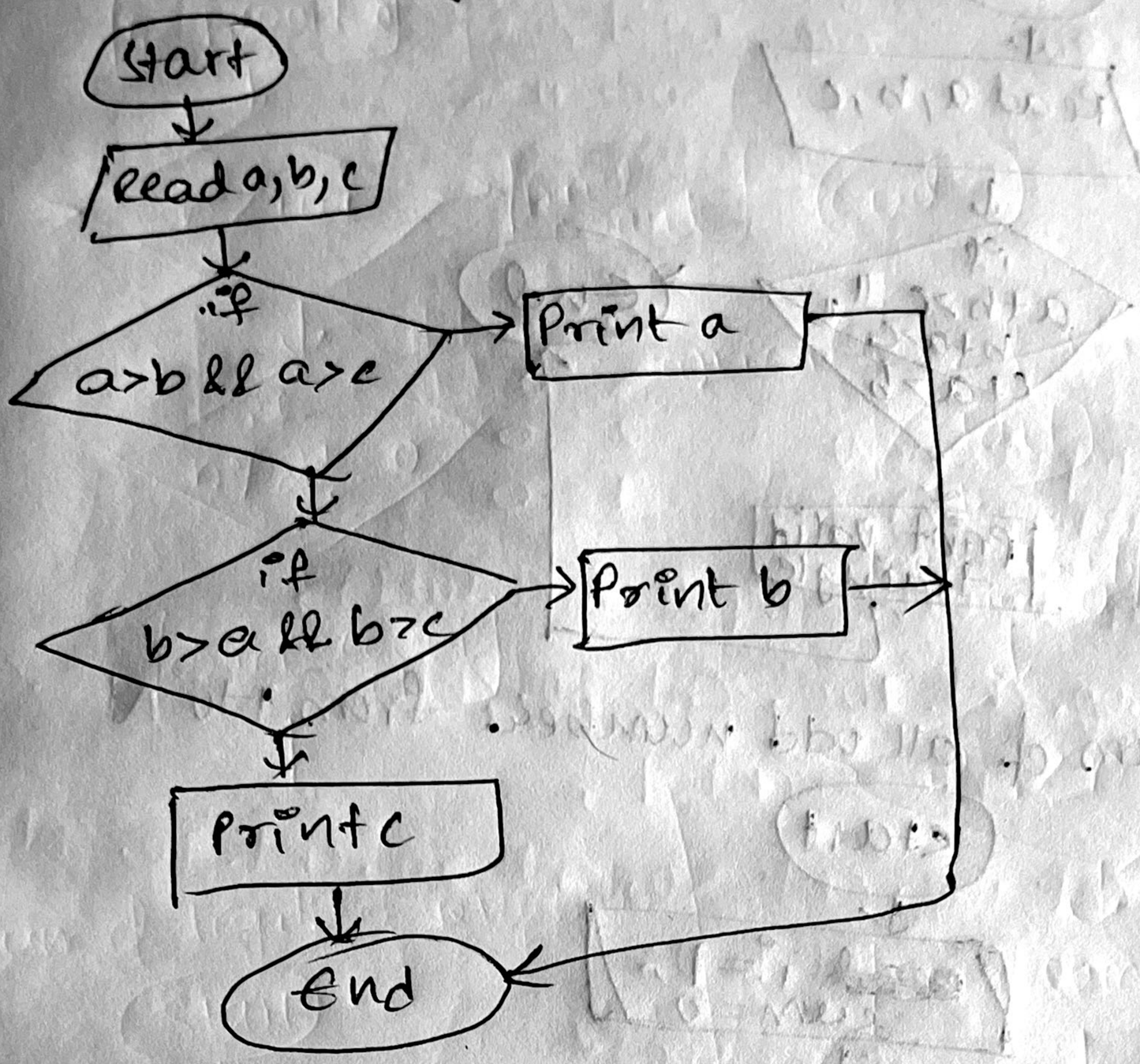






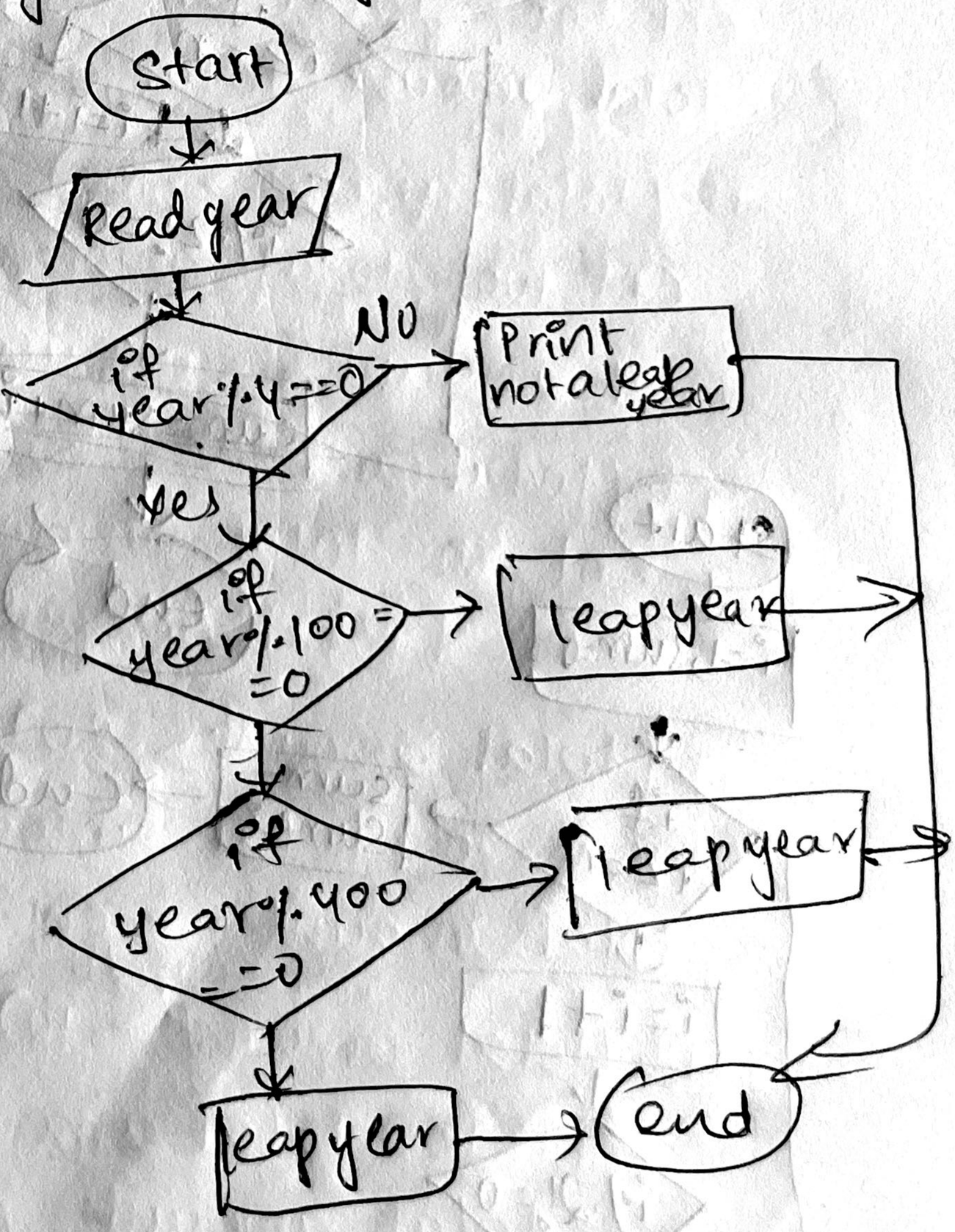
Flowchart

1) find max of 3 given integers

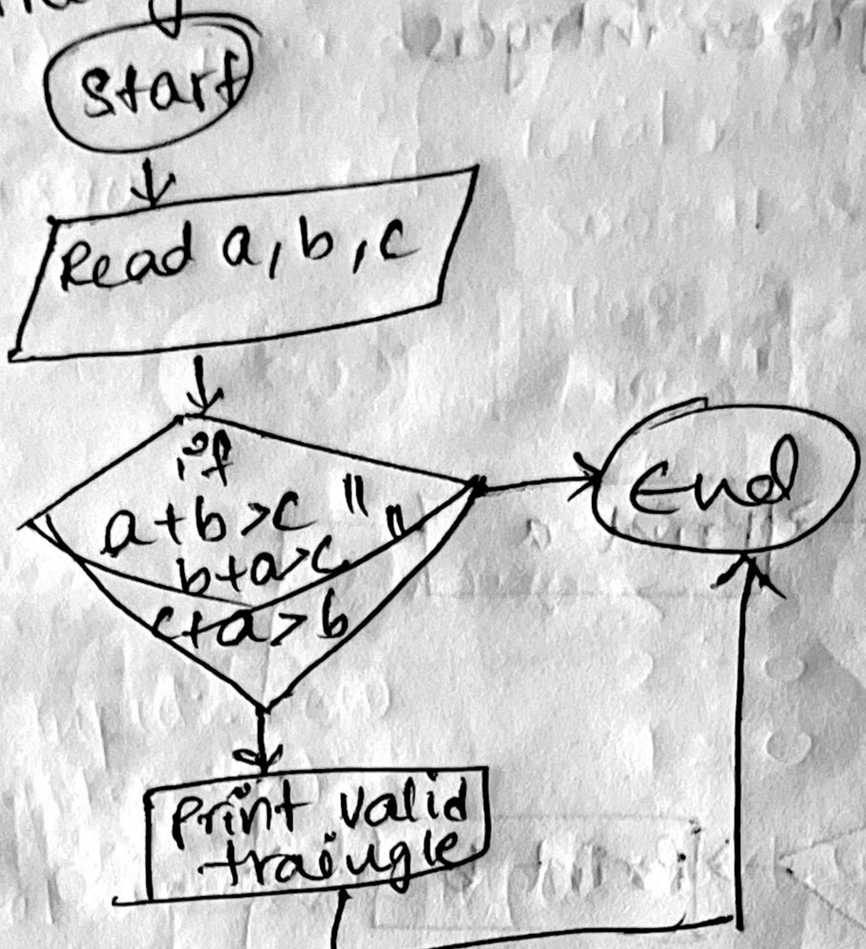


2) whether a given year is leap or not?

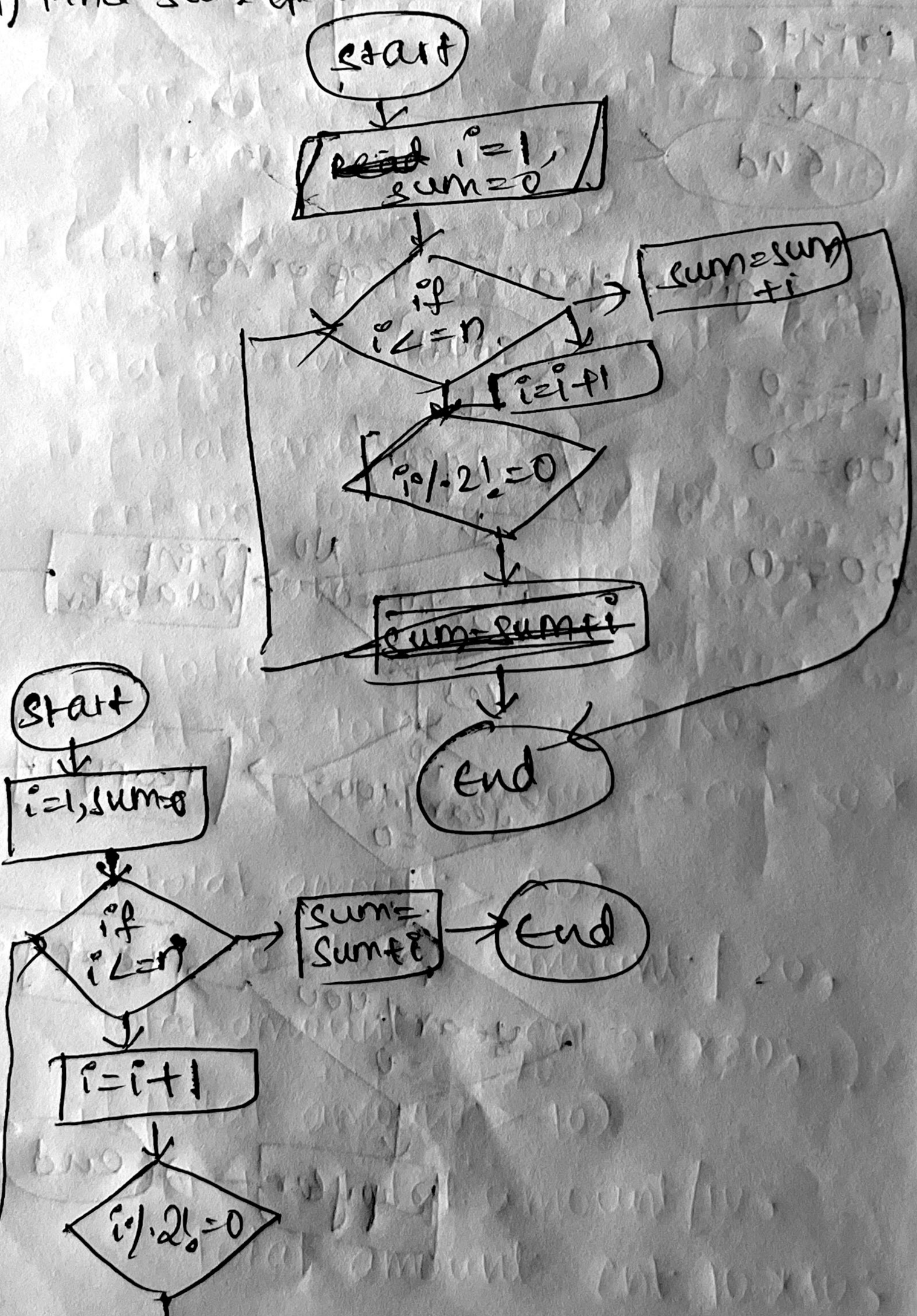
① if n%.4==0
n%.100==0
n%.200==0



3) Given 3 sides of a ... valid traingle or not?



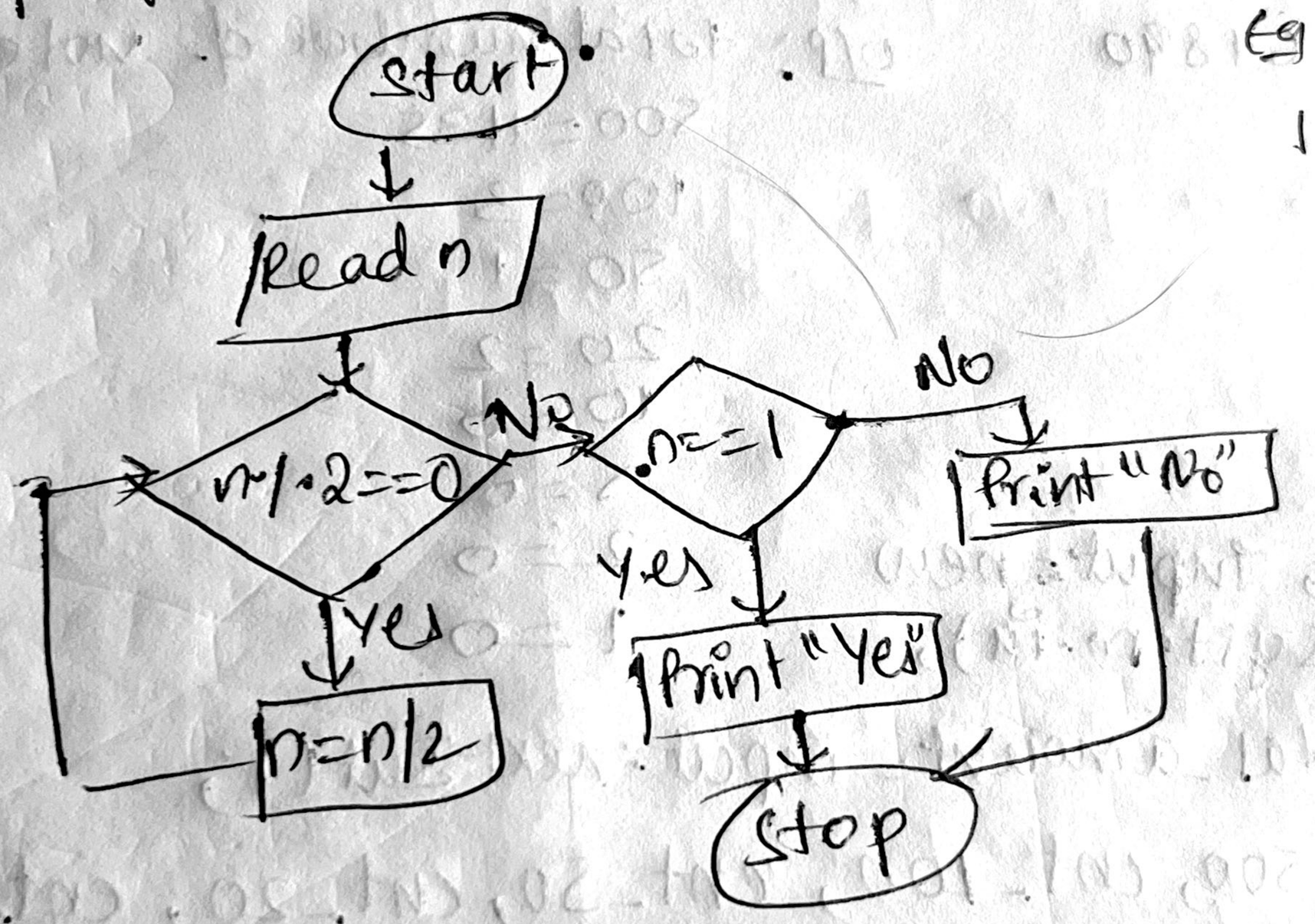
4) Find sum of all odd numbers from 1 to N



5. Find if a given number is a perfect no of 2 or not?.

Eg: 1, 2, 4, 8, 16, ... etc

are perfect powers of 2, whereas 3, 6, 10



Eg: 16

16%2==0

↓

16/2=8

↓

8%2==0

↓

8/2=4

↓

4%2==0

↓

4/2=2

↓

2%2==0

↓

2/2=1

↓

∴ Power of 2