

* Data Structures and Algorithms *
(C++)

Basics to Advance

LinkedIn Profile: subrat-kumar-singh-597973207

Written By: Subrat Kumar Singh.

Contact NO: 9350857818.

gmail: subratsingh2001@gmail.com

Date
27/01/2023

* Data Structure and Algorithms *

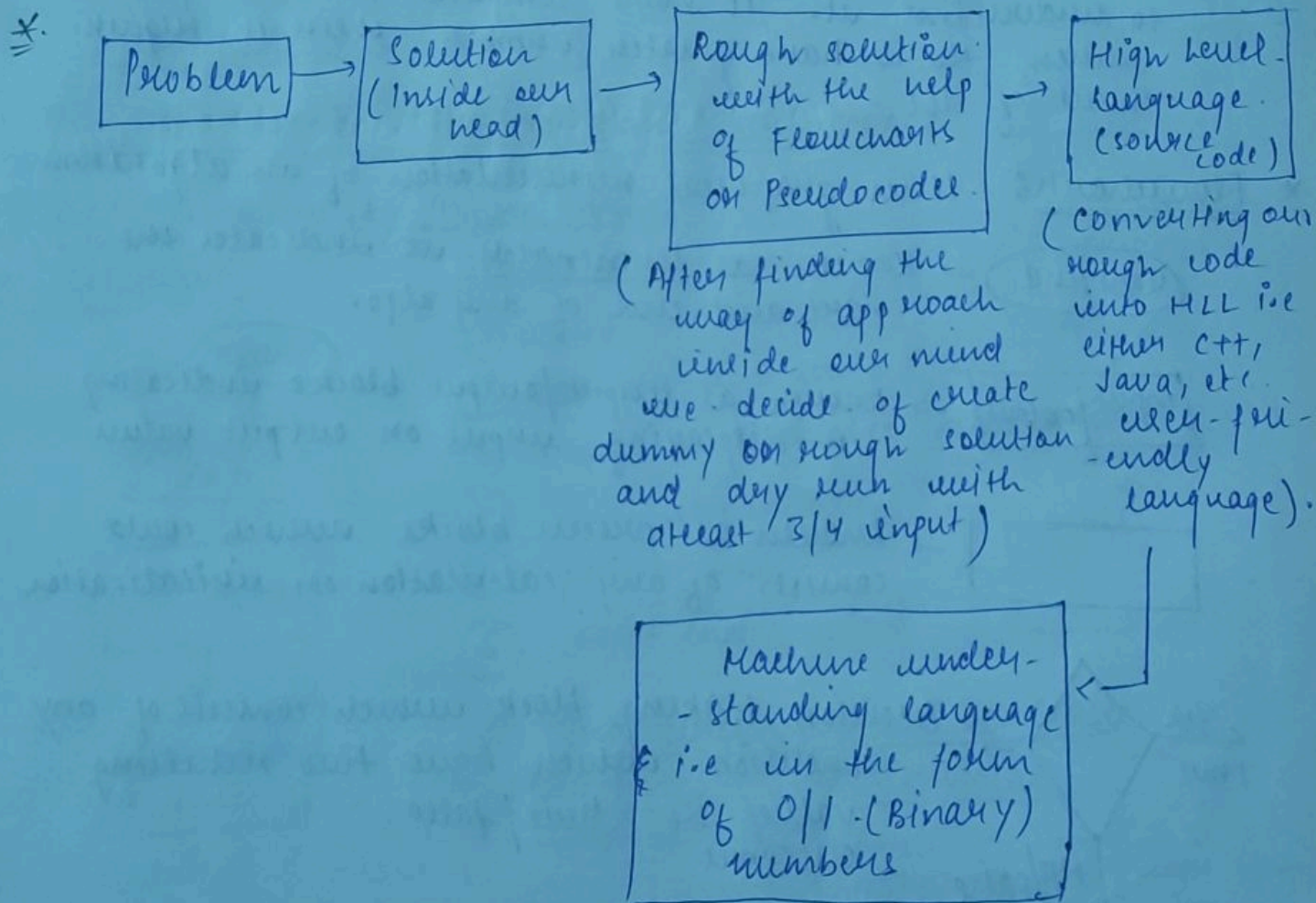
C++

* Programming Fundamentals:

→ Thought Process to solve a Problem:

- * Understand the problem.
- * Look out for input/output values.
- * find out for best approach i.e algorithm to solve that particular problem.

* Algorithm: steps involved in accomplishing a task known to be algorithm.



* Between HLL and Machine understanding language there is an intermediate known as Compiler (it converts HLL into Machine understanding language i.e form of 0/1 because machine won't be able to understand HLL can understand only 0 and 1 so compiler converts form of 0 and 1).

* Using computer to solve a problem:

Consider an example to check whether a number is prime or not. ex: num = 13

Prime nos are those are divisible by 1 and the no itself.

let check:

$$13 \% 1 = 0$$

$$13 \% 2 = 1$$

$$13 \% 3 = 1$$

$$13 \% 4 = 1$$

$$13 \% 5 = 3$$

⋮

$$13 \% 13 = 0$$


So 13 is divisible by 13 and 1
as rem. is 0. \therefore it is prime

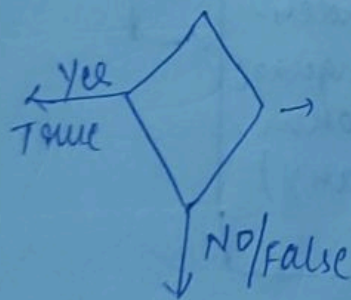
* But the approach is manually which computer is unable to understand it. To make computer solve the problem when we have greater numbers, there is requirement of HLL.

* Flowcharts: It is graphical representation of an algorithm.

start/end \rightarrow Known as terminator. it indicates the start and end of any algo.

input/output \rightarrow known as input/output blocks indicating in displaying input or output values.

 \rightarrow known as process blocks which could consists of any calculation or initialization.



Decision Making block. which consists of any condition which have two resulting value/ i.e. true/false outcomes.



\rightarrow Another Flow of execution of an Algo.



: Connector indicates the function.

* Pseudo Code: Generic way of implementation of algo in the textual form.

ex: ① To add 2 numbers

- ① start
- ② Take input a and b from user
- ③ $sum = a + b$
- ④ print sum
- ⑤ exist end

* Pseudo code can vary from person to person.

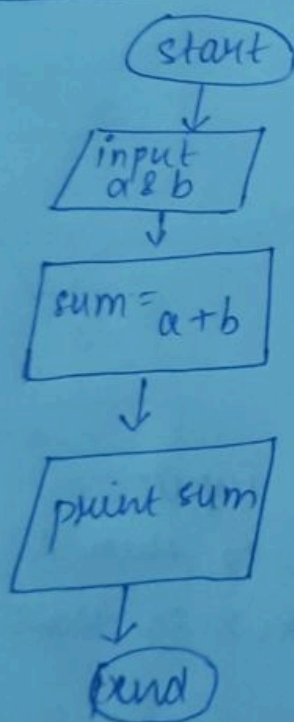
ex: ② To multiply 2 numbers

- ① start
- ② Take input a and b from user.
- ③ $multiply = a \times b$
- ④ print multiply
- ⑤ exist end

* Time to Practice Flowcharts and Pseudo codes:

① Add 2 numbers by taking Input:

* Flowchart

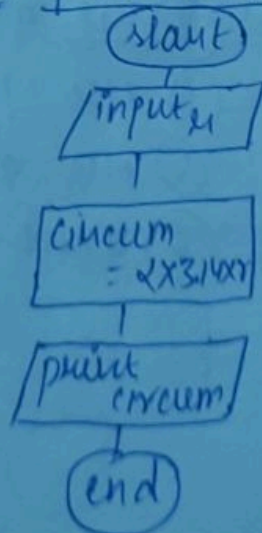


* Pseudocode

- ① start
- ② Take input a and b from user
- ③ create a variable named sum.
- ④ Then $sum = a + b$
- ⑤ print sum
- ⑥ exist end

② Find circumference of Circle

* flowchart

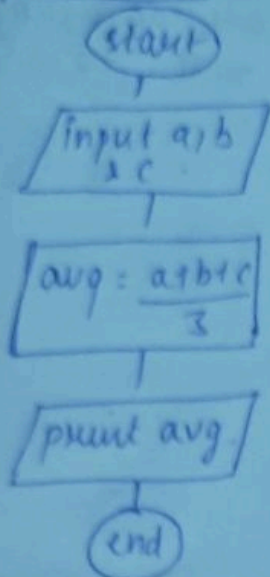


* Pseudocode

- ① start
- ② Take input r from user
- ③ create a circum variable
- ④ $circum = 2 \times 3.14 \times r$
- ⑤ print circum
- ⑥ exist end

③ Average of 3 numbers:

* Flowchart

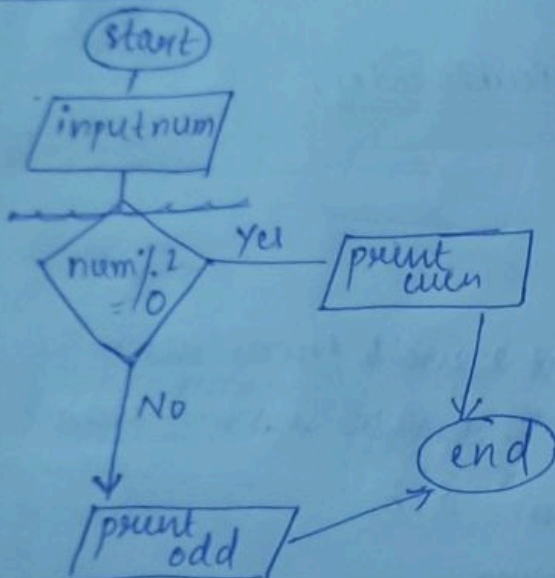


* Pseudocode

- ① start
- ② Take input a, b, and c from the user
- ③ create avg a variable
- ④ $avg = \frac{a+b+c}{3}$
- ⑤ print avg.
- ⑥ exist end

④ check number is odd or even:

* Flowchart

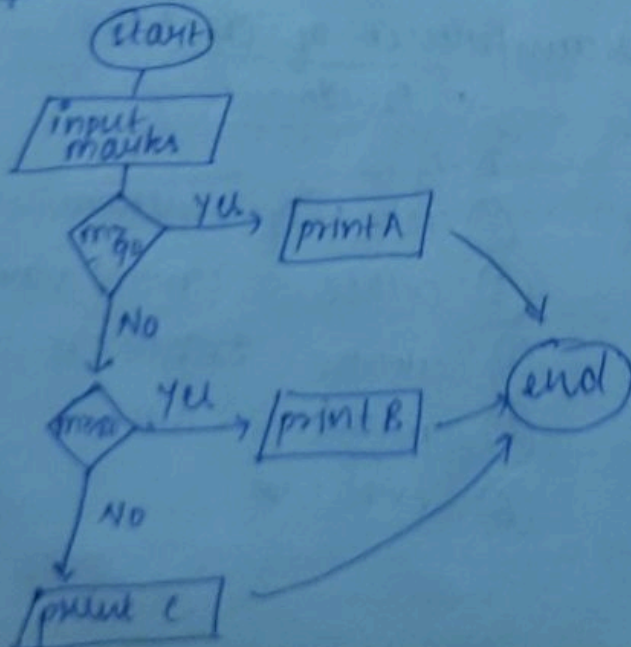


* Pseudocode

- ① start
- ② take input num.
- ③ if $num \% 2 = 0$ then print even
- ④ else print odd
- ⑤ exist end

⑤ Students and Grade Flowchart:

* Flowchart

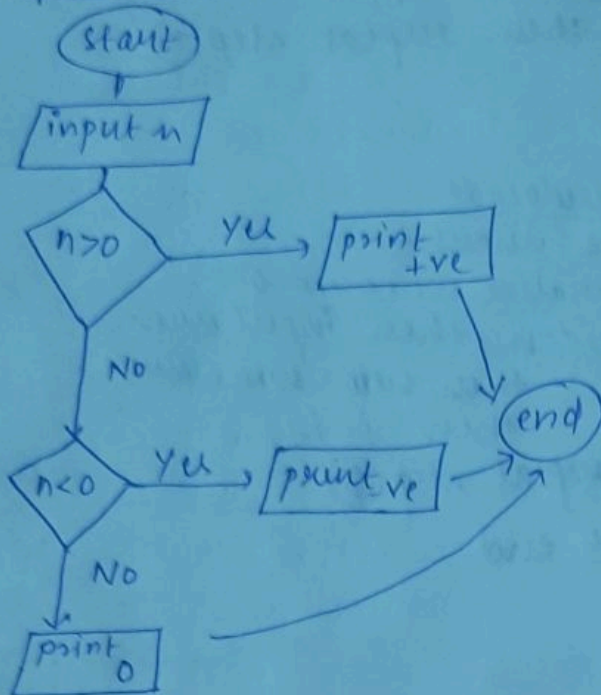


* Pseudocode

- ① start
- ② take marks input
- ③ if marks ≥ 90 then A.
- ④ else if marks ≥ 80 then B
- ⑤ else grade C
- ⑥ exist end

⑥ Check number is +ve, -ve or 0.

* flowchart

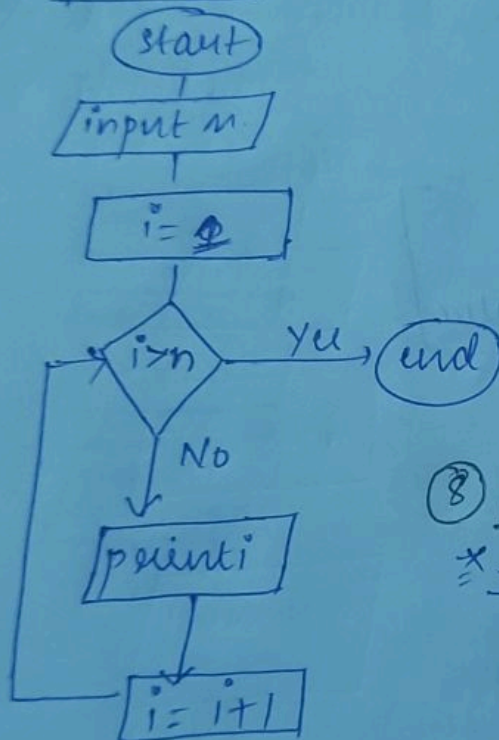


* Pseudocode

- ① start
- ② take input n from user
- ③ if $n > 0$, then +ve
- ④ else if $n < 0$, then -ve
- ⑤ else 0
- ⑥ exist end.

⑦ Print counting from 1 to N.

* flowchart

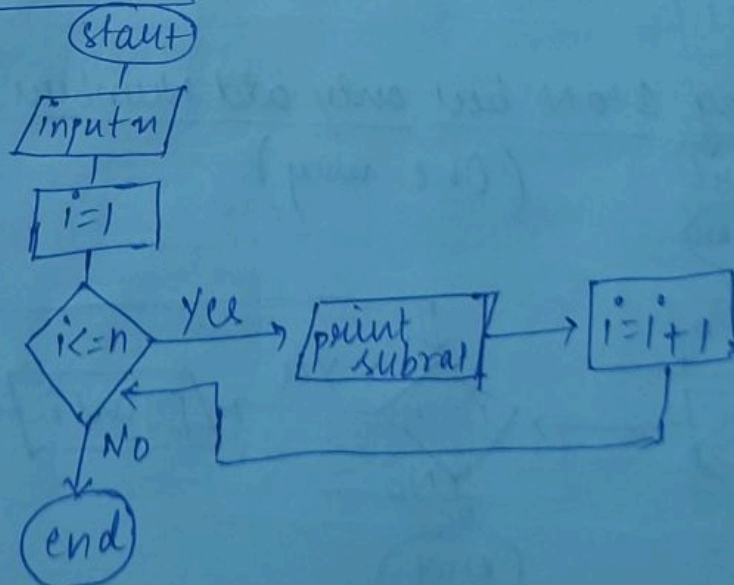


* Pseudocode

- ① start
- ② take input n from user.
- ③ take $i = 1$.
- ④ if $i > n$, then end
- ⑤ else print i, then $i = i + 1$
then repeat step 3
- ⑥ ~~end~~ end.

⑧ printing Your Name n-times

* flowchart



* Pseudocode

① start

② input n

③ take $i = 1$

④ if $i \leq n$, then print subrat

⑤

then $i = i + 1$

then repeat step ③

⑥ else end

⑨ Add N numbers from user:

ex: Ist no: 3

IInd no: 4

IIIrd no: 5

sum = 12

* Pseudocode

① Take input n

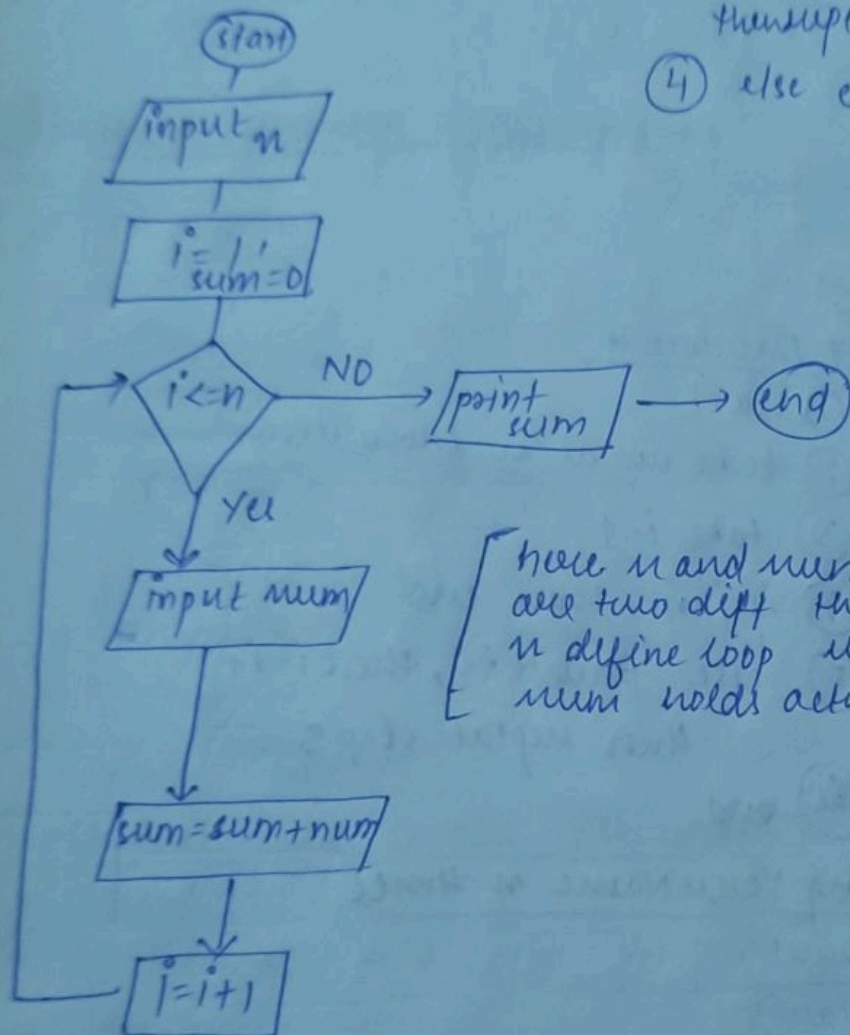
② Initialize $i = 1$, $sum = 0$

③ if $i \leq n$, then input num
then $sum = sum + num$
then $i = i + 1$

then repeat step ②

④ else end

* flowchart

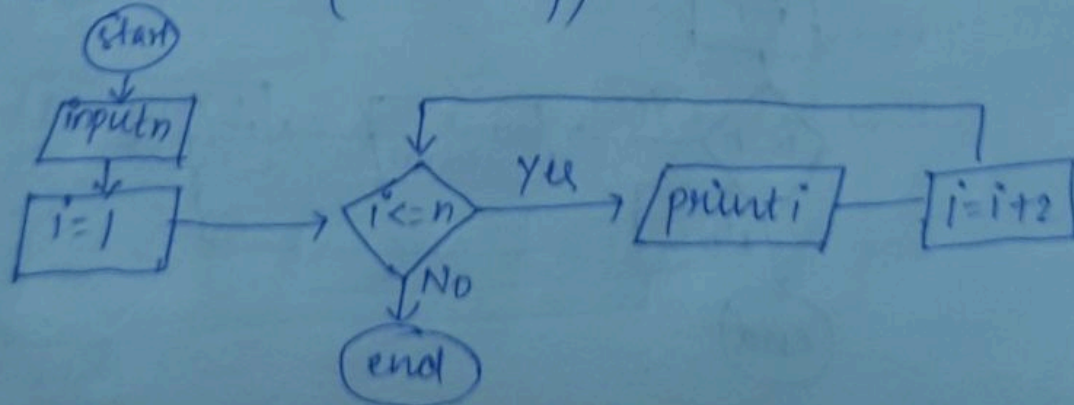


here n and num
are two diff things
 n define loop where
 num holds actual value

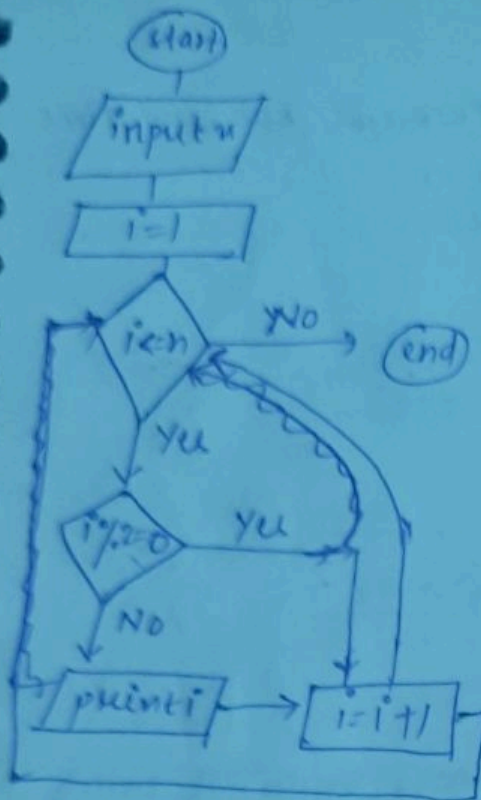
⑩ Printing 1 to N but only odd Numbers:

* flowchart

(One-way)



Second-way

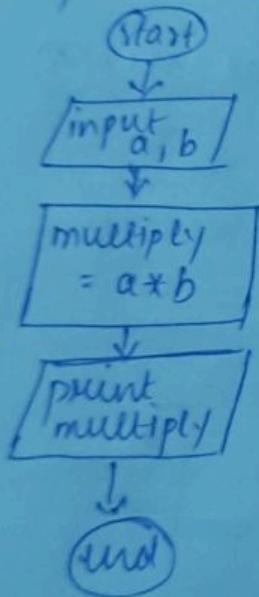


* Pseudocode

- ① start
- ② Take input n .
- ③ initialize $i = 1$ check
- ④ if $i \leq n$, then $i \% 2 = 0$
- ⑤ else end
- ⑥ if $i \% 2 = 0$, then $i = i + 1$, then repeat step ④
- ⑦ else print i , then $i = i + 1$ then repeat step ④
- ⑧ ~~exit~~ end

⑪ Multiply two no by taking input

* flowchart

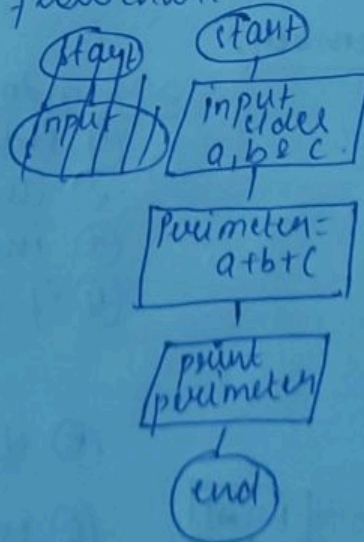


* Pseudocode

- ① start
- ② take input a and b
- ③ multiply = $a * b$
- ④ print multiply.
- ⑤ end.

⑫ Find the perimeter of a Δ?

* flowchart

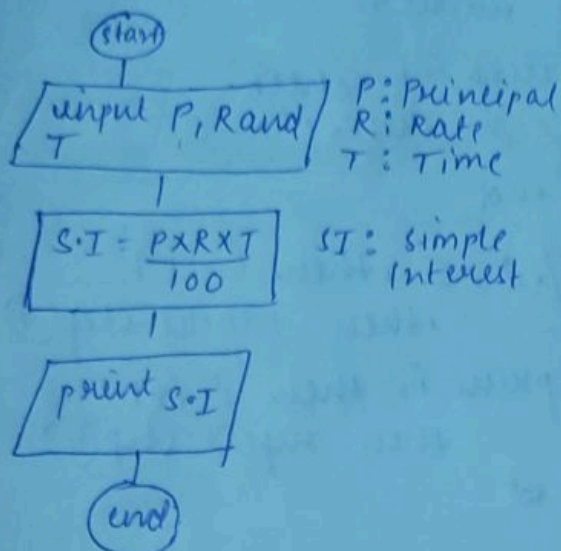


Pseudocode

- ① start
- ② input sides a, b and c
- ③ Perimeter = $a + b + c$
- ④ print perimeter
- ⑤ end.

13) Find simple interest:

* flowchart

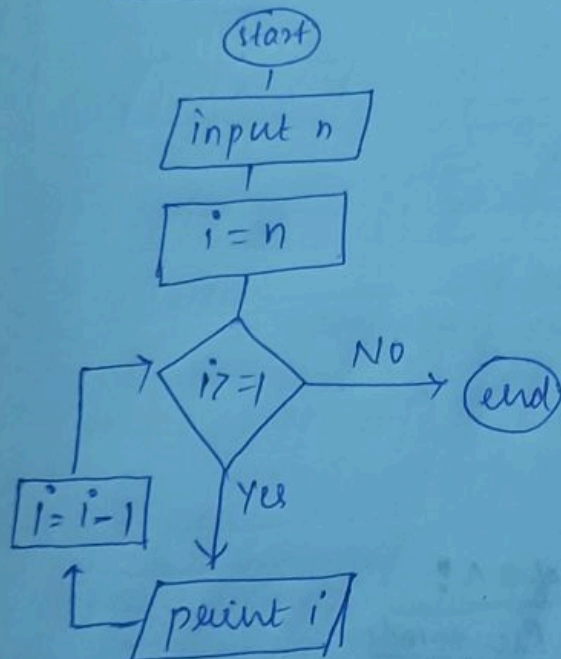


* Pseudocode

- ① start
- ② input Principal, Rate and Time.
- ③ $S.I = \frac{P \times R \times T}{100}$
- ④ print S.I
- ⑤ end.

14) Print counting from N to 1:

* flowchart

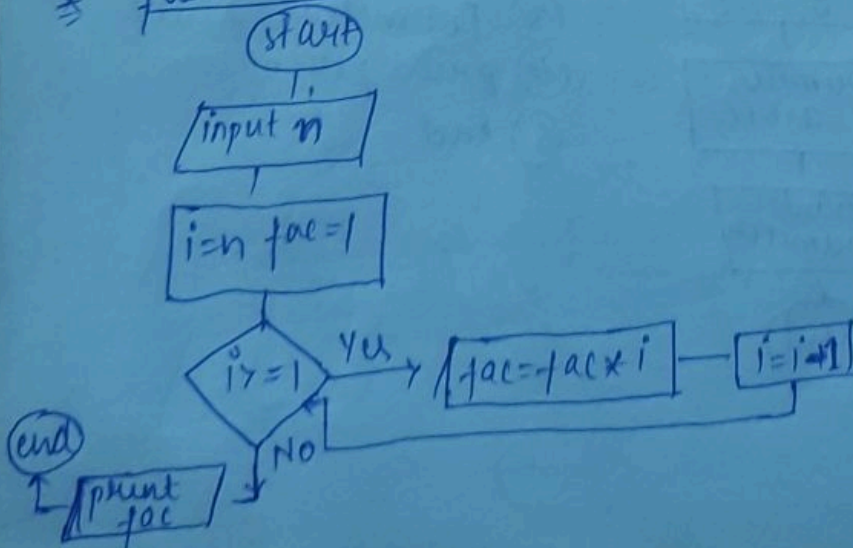


* pseudocode

- ① start
- ② take input n.
- ③ initialize i = n
- ④ if $i \geq 1$, then print i,
then $i = i - 1$
then repeat step ④
- ⑤ else end

15) Find Factorial of a Number:

* flowchart

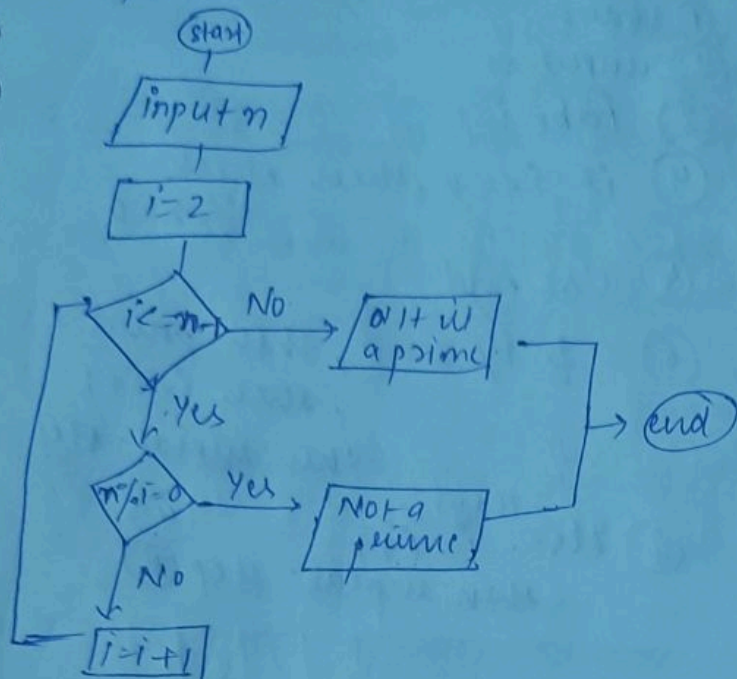


* Pseudocode

- ① start
- ② input n
- ③ initialize i = n, fac = 1
- ④ if $i \geq 1$, then $fac = fac \times i$
then $i = i - 1$
then repeat step ④
- ⑤ else print fac
- ⑥ end.

16) Check no is prime or not?

* flowchart



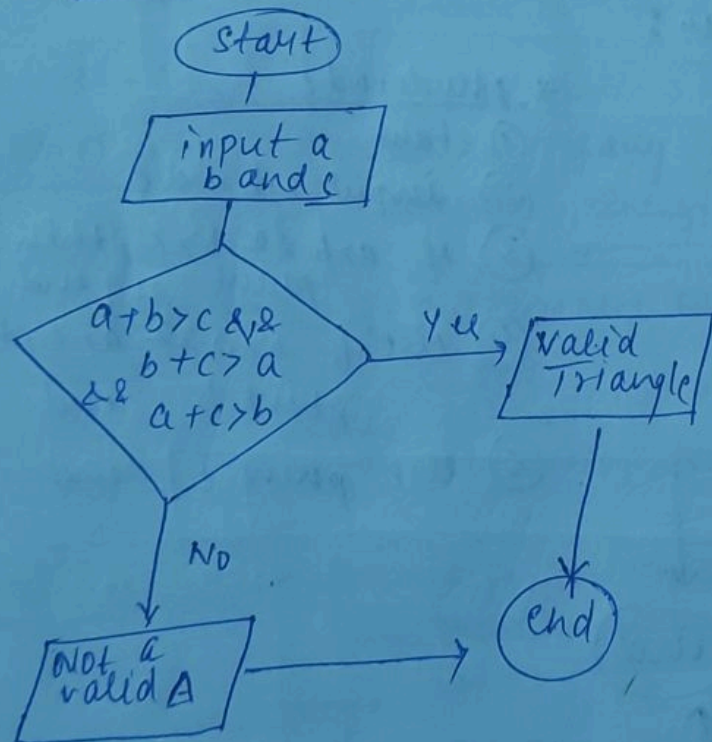
Pseudocode

- ① start
- ② take input n
- ③ take $i = 2$
- ④ Now if $i \leq n-1$, then check $n \% i = 0$
- ⑤ else ~~no~~ it is a prime then end
- ⑥ If $n \% i = 0$ then Not a prime then end
- ⑦ else $i = i + 1$ then repeat step ④

17) Valid Triangle or not?

$a+b > c$, $b+c > a$ and $a+c > b$

* flowchart

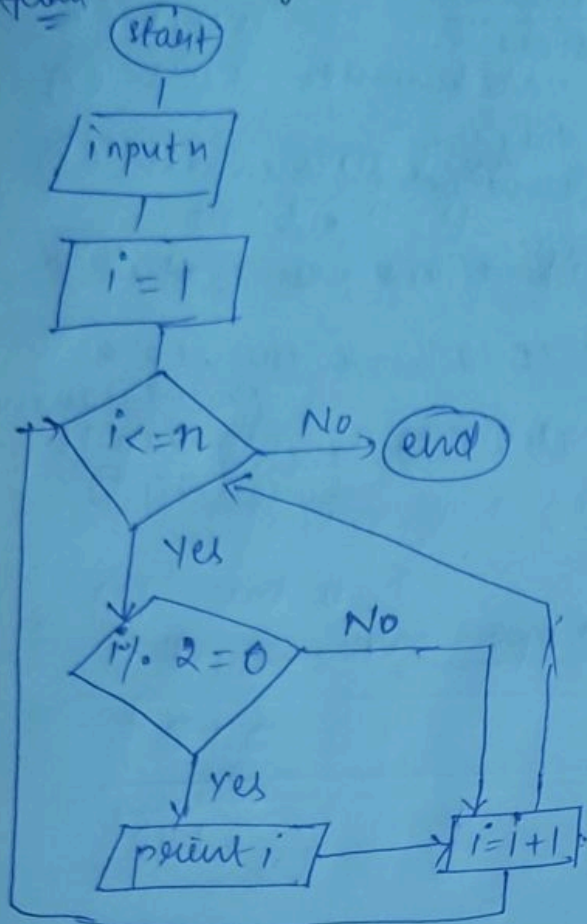


* Pseudocode

- ① start
- ② input a, b and c
- ③ if $a+b > c$ and $b+c > a$ and $a+c > b$ then called valid Δ 's
- ④ else not a valid Δ
- ⑤ end

18) Printing 1 to N but only even numbers:

Flowchart

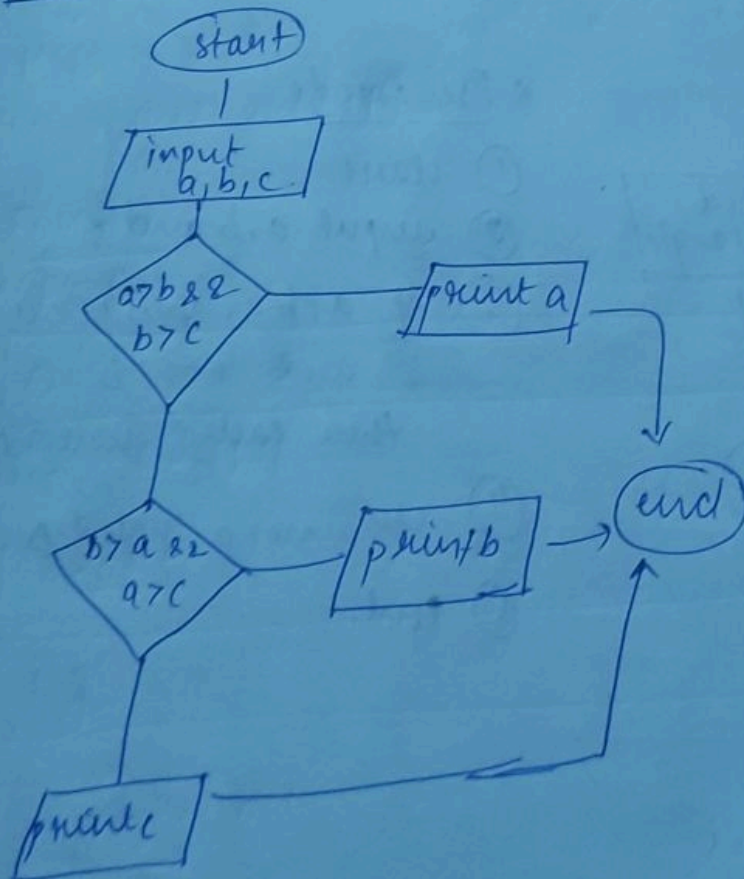


* Pseudocode

- ① start
- ② input n
- ③ take $i = 1$
- ④ if $i \leq n$, then check $i \% 2 = 0$
- ⑤ else end
- ⑥ if $i \% 2 = 0$, then print i, then $i = i + 1$, then repeat step ④
- ⑦ else. $i = i + 1$, then repeat step ④

19) Print max of three numbers:

* flowchart



* pseudocode

- ① start
- ② input a, b and c
- ③ if $a > b$ & $b > c$, then print a end
- ④ else if $b > a$ & $a > c$, then print b end
- ⑤ else print c end