

Date : 21/08/23

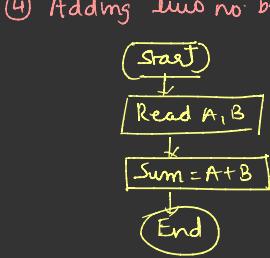
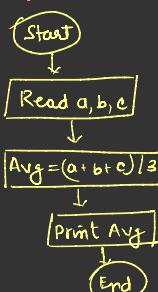
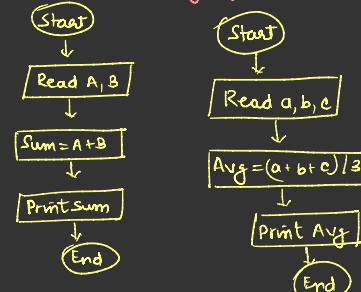
## -- Orientation Class --

### SCHEDULE

- Monday : 9 PM
- Wednesday : 9 PM
- Friday : 9 PM (recorded)
- Sunday : 9 PM ~ Lakshay

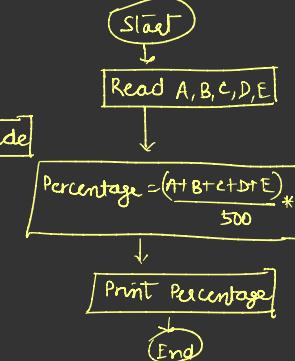
- MCQ
- Debugging exercises
- Cheatsheet

### ① Sum of A,B : ② Avg of a,b,c : ④ Adding Two no by taking Input

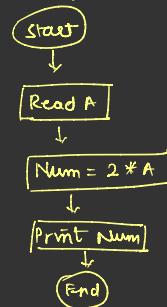


### ⑥ Percentage of Overall Marks

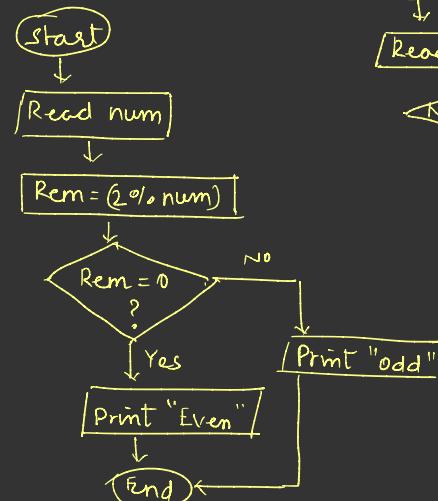
- (A) → Eng : 88  
 (B) → IP : 99  
 (C) → Phy : 95  
 (D) → Chem : 95  
 (E) → Math : 95



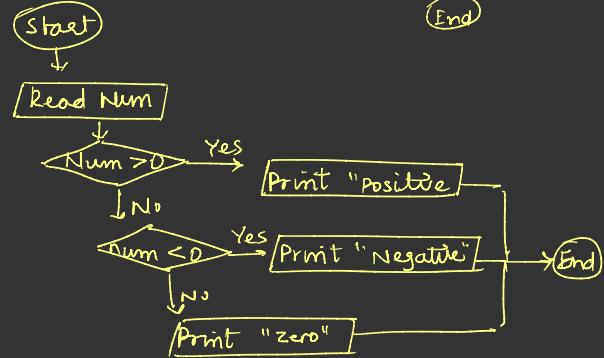
### ③ Print twice of A



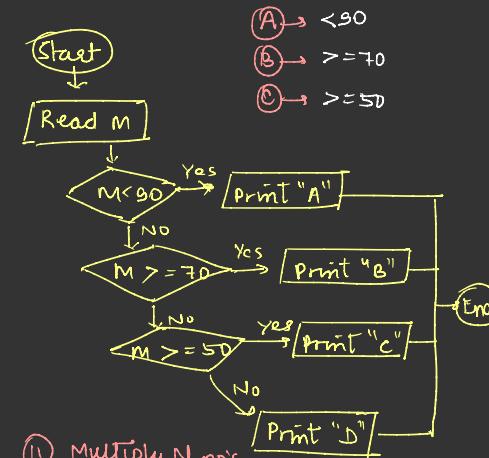
### ⑦ Even or Odd



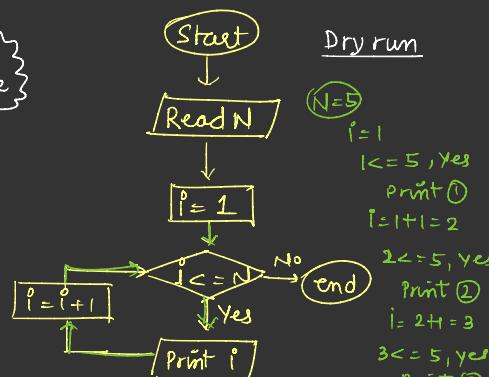
### ⑧ Positive / Negative / 0



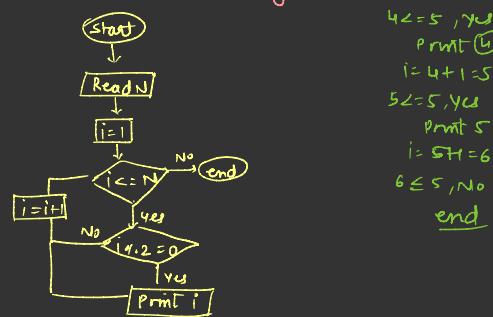
### ⑨ Student grade chart



### ⑩ Counting from I to N.

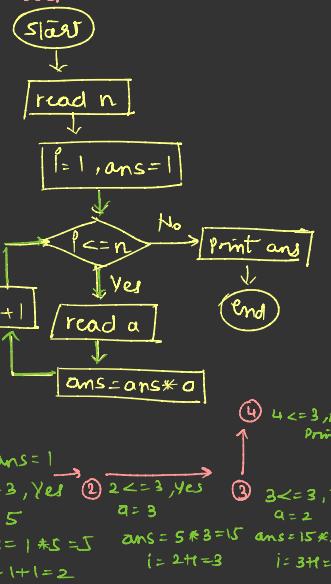


### ⑪ Print 1 to N, but even only



Dry run  
 $N=5$   
 $i=1$   
 $i \leq 5, Yes$   
 $i=1+1=2$   
 $2 \leq 5, Yes$   
 $i=2+1=3$   
 $3 \leq 5, Yes$   
 $i=3+1=4$   
 $4 \leq 5, Yes$   
 $i=4+1=5$   
 $5 \leq 5, Yes$   
 $i=5+1=6$   
 $6 \leq 5, No$   
 $end$

### ⑫ Multiply N no's from User



④  $4 \leq 3, No, print 30$   
 $i=1, ans=1$   
 ①  $1 \leq 3, Yes$   
 $a=5$   
 $ans = 1 * 5 = 5$   
 $i = 1 + 1 = 2$   
 ②  $2 \leq 3, Yes$   
 $a=3$   
 $ans = 5 * 3 = 15$   
 $i = 2 + 1 = 3$   
 ③  $3 \leq 3, Yes$   
 $a=2$   
 $ans = 15 * 2 = 30$   
 $i = 3 + 1 = 4$

### Flowchart Components

(Start/end) or (start) : Terminator

Read a,b : Read / Print

sum = a+b : Process Block

Decision Making block

④ : Connector

{functions}

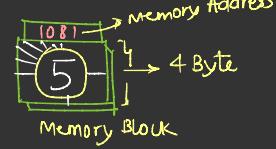
Date - 25 Aug 23

-- Write your first C++ program -

## (Datatypes and Variables)

Type of Data  
Name of Memory Location

`[int] [Number] = [5];`  
Datatype      Variable      value



Datatype

Type of Data + Size of Memory

[Integer]: -21, -47, ..., 0, 1, 2, 97, 88, ...

int num; → declaration  
int num = 55; → Initialisation

[sizeof(num)] → Checks size of num

[64bit CPU]

① int - 4 Byte

② long - 8 Byte

③ char - 1 Byte

④ short - 2 Byte

⑤ long long - 8 Byte

⑥ float - 4 Byte

⑦ double - 8 Byte

→ If there is no definition then a garbage value will be stored.

## Naming Conventions

int a = 20;

int totalStudents = 20;

## How data is stored?

① int number = 7;

int → 32 Bits (4 Byte)

00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000  
1 Byte      number

int num2 = 10;

In Binary: 010

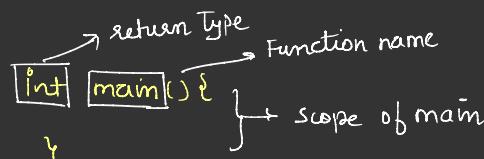
2<sup>3</sup> 0 5  
2<sup>2</sup> 1 0  
2<sup>1</sup> 0 1  
2<sup>0</sup> 1 0  
00000010

② char ch = 'a'; → ASCII Value

ch → 97 = Binary of 97  
11110111

③ bool flag = true;  
1 Byte → true  
But 1 bit was sufficient  
Because:  
Minimum Addressable Memory = 1 Byte

Minimum Addressable Memory = 1 Byte



Preprocessor Directive

#include <iostream> → Definition of cout

Using namespace std; → namespace info

int main() {

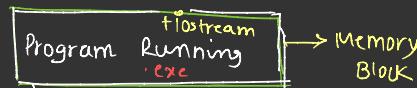
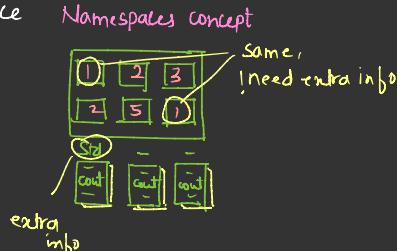
cout << "Love Babbar";

print syntax (Insertion operator)

}

Executed:

- Before anything include — header file
- Running program : Process
- 



→ Preprocessor Directive

Before processing the program  
PD includes the files.

endl :  
'\n' :  
→ prints in a new line  
new line character

① #include <iostream> → Preprocessor directive  
 ② Using namespace std; → Namespace : Standard  
 ③ int main() → Entry point  
 ④ {  
 ⑤ cout << "Babbar"; → Prints "Babbar".  
 ⑥ return 0;  
 ⑦ } → { } : Scope of main  
 ; → return 0: successful execution  
 ; → Termination of line

cin → take Input From User

int marks  
cin > marks;

Saves the entered in marks variable

How negative's stored?

## Operators

$0000011 \rightarrow 111100$   
 $10101100 \rightarrow 01010000$   
 $11111100 \rightarrow 00000011$   
 $10111001 \rightarrow 01000110$   
 $11100111 \rightarrow 00011000$

- ① Arithmetic  $\rightarrow (+, -, /, *, \% )$
- ② Relational  $\rightarrow (>, <, >=, <=, !=, ==)$
- ③ Assignment  $\rightarrow (=)$
- ④ Logical  $\rightarrow (\text{AND}, \text{OR}, \text{!})$
- ⑤ Bitwise  $\rightarrow$ 
  - $\text{AND} \rightarrow 0 \rightarrow 1$
  - $\text{OR} \rightarrow 1 \rightarrow 0$
  - $\text{Non-zero} \rightarrow 0$

2's complement  $\rightarrow$  1's complement + 1

$$\begin{array}{r} 1011 \rightarrow 0100 \\ \quad +1 \\ \boxed{0101} \end{array}$$

$$\begin{array}{r} 1000 \rightarrow 0111 \\ \quad +1 \\ \hline 0100 \end{array}$$

Negative no's  $\rightarrow$  Ignore -ve sign

- $\rightarrow$  find Binary
- $\rightarrow$  take 2's complement
- $\rightarrow$  store

int a = -5

$\rightarrow$  5

$\rightarrow$  Binary : 00000000 00000000 00000000 00000101

$\rightarrow$  1's complement: 11111111 11111111 11111111 11110100

2's complement:



① char  $\rightarrow$  1 Byte : 8 bits

Signed

② long  $\rightarrow$  8 Byte  $\rightarrow$  64 Bits

01101010101010101010101010101010

Possible combination:  $2^8 \rightarrow 256$

Possible combs:  $2^{64}$

Range:  $[2^8 - 1]$   $0 \rightarrow 255$

Range  $\rightarrow 2^n - 1$

Generic formula:  $[2^n - 1]$

=

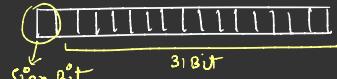
Signed and unsigned data

+ve, 0, -ve

+ve

unsigned

1Bit



0  $\rightarrow$  +ve  
1  $\rightarrow$  -ve

① char ch = 'a';



$2^7 \rightarrow [2^7 - 1]$

$= 128 - 1$   
 $= 127$

Range:-

$-2^7 \rightarrow 2^7 - 1$

$[-128 \rightarrow 127]$

② int

$0 \rightarrow 2^{32} \rightarrow$  Range

Unsigned

$[-----| | | |]$

Signed

$[-----| | | |]$

$[-2^{31} \rightarrow 2^{31}] \rightarrow$  Range