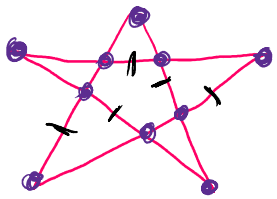
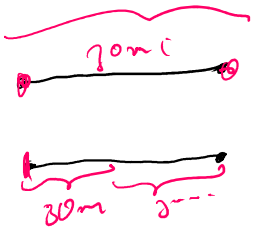
Foundation Class - 3

Sunday, August 17, 2025 2:52 PM

Puzzles

Q) 10 balls, 5 lines — Every line should have 4 balls.



Q) 

Two wires that burn in 1 hour each.

⇒ measure 45 mins using the wires.

Q) 50 R, 50 Blue balls

Pulling out a red ball Prob. max

$\frac{1}{2}(1) + \frac{1}{2}0 = \frac{1}{2}$

$\frac{1}{2} \times \frac{25}{75} + \frac{1}{2}(1)$

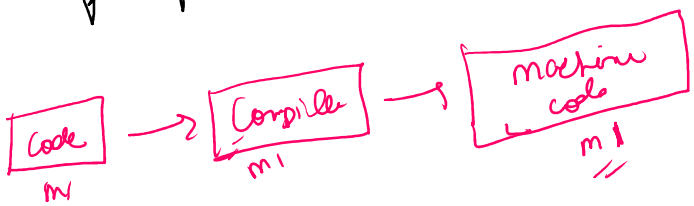
$= \frac{1}{6} + \frac{1}{2}$

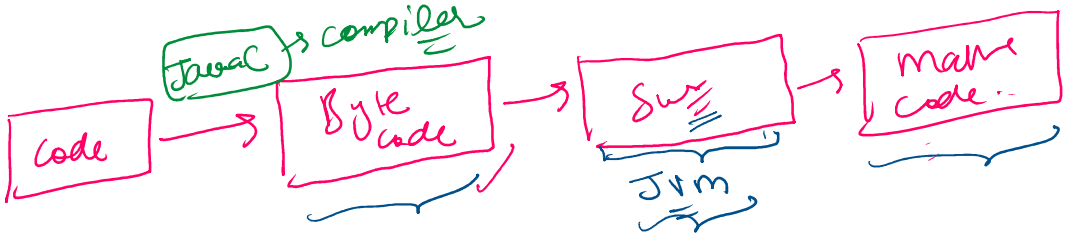
$\frac{1}{2} \times \frac{49}{99} + \frac{1}{2}(1) = 0.7474$

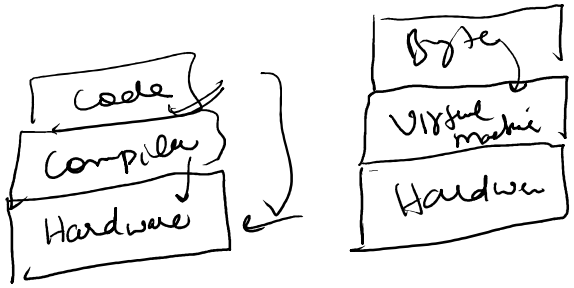
Java Architecture

- JDK] → Java Development Kit
- JRE] → Java Runtime Env
- JVM] → Java Virtual machine
- JIT] → Just In Time Compilation

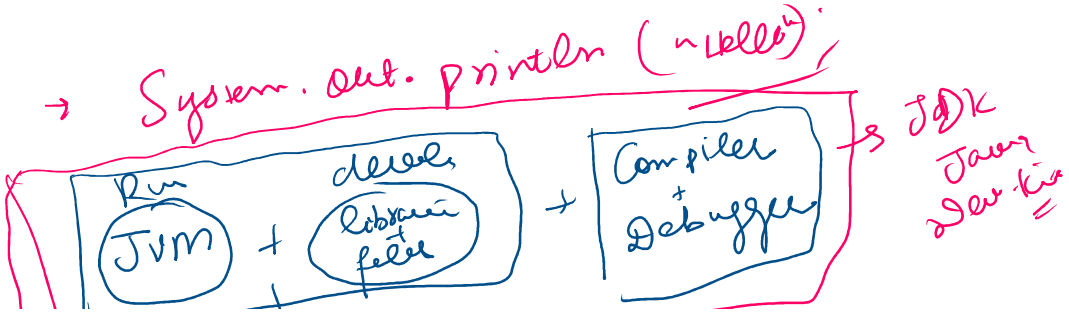
(A) Java is platform independent

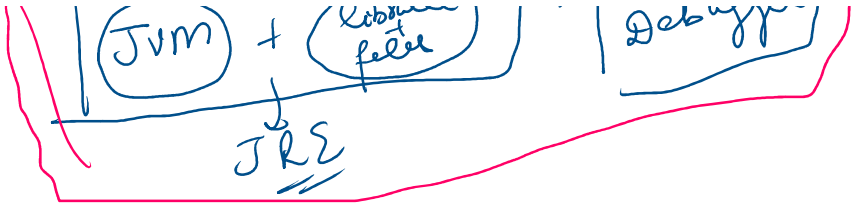






JVM] → Env which helps you run the Bytecode





JIT : Just In time compilation

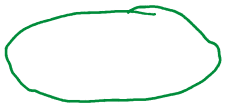
⇒ Principle of locality | Locality of Reference

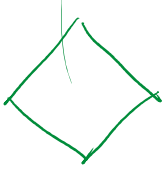
A processor accesses 10% of the memory location for 90% of the times.

Flow charts

↓

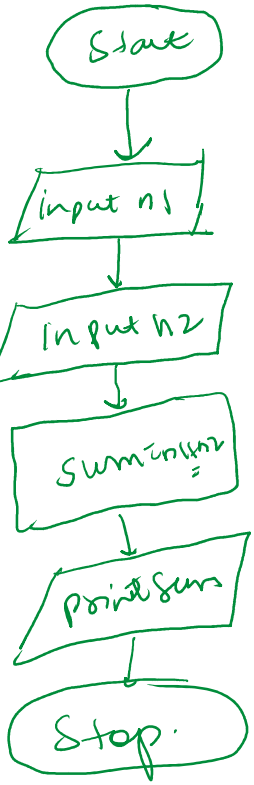
pictorial representation of an algo



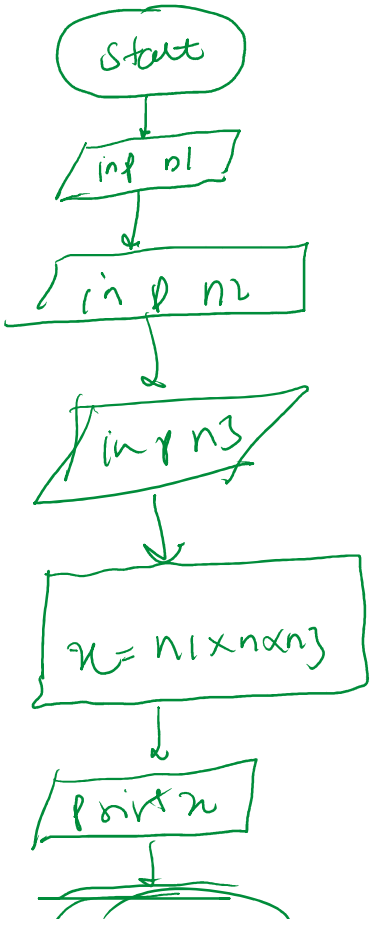


Q) Add two numbers

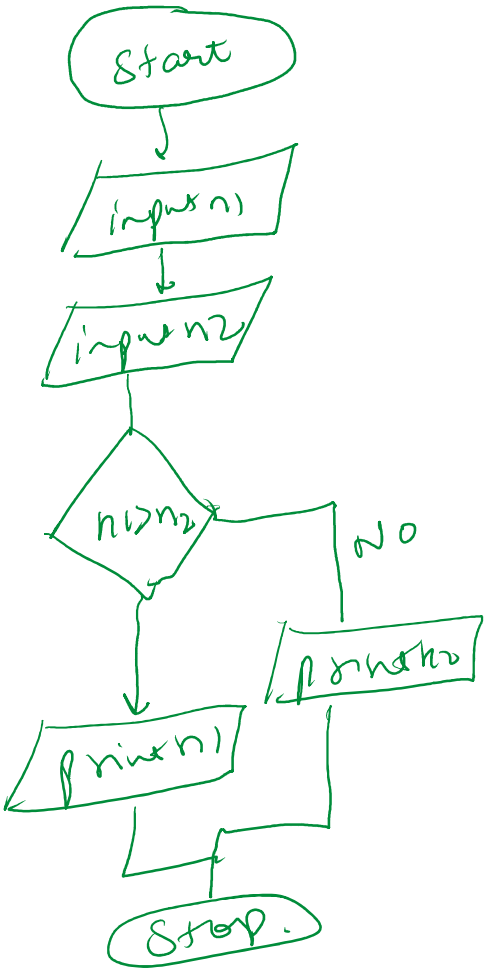
8) Add two numbers



9) Draw a flowchart to multiply 3 nos.



0) flow chart to find bigger of the two numbers.



Pseudo code

↳ false / Idea

→ Step by Step representation of algorithmical algo

↳ Pseudo code is not written in any programming language.

→ Pseudo code is not a programming language.

A good Pseudo code can be converted into original code in any language.

→ We can use IF, IF-Else statements in Pseudo code

Q) Pseudo code for adding two nos,

1) Take n1 as input
2) Take n2 as input
3) Add n1 and n2 and store in a variable (sum)
4) print sum.

Q2) Pseudo code to find the square of number

→ Start
  Input n;
  Store n×n in ans;
  print ans;
End

[IF-
IF-Else

→ Pseudo code to find the square

0) Pseudo code to find the larger of the two numbers:

```
Start
  Input n1, n2
  IF n1 > n2
     print n1
  ELSE
     print n2
End
```

0) Pseudo code to find factorial of a number

Sol:

```
Start
  Input n;
  Initialize fact = 1
  Run for i from 1 → n;
      fact = fact × i;
  print fact
End
```

ee

Arrays → 1D
→ 2D

Data Structure

#

- Arrays → 1D → 2D
- Array list
- Strings

Data Structures

- Linear
  - Array, Array list, Linked list, Stacks, Queues
- Non-linear
  - Trees, Graphs

Advanced Data Structures

- DSU: Disjoint Set Union
- Segment Trees
- Fenwick Trees
- Convex Hull
- Trie

Advanced Algos

- Multi Dimensional DP
- Game Theory

Basics of Java

- print

```
cout << "Hello World";
```

end of a program statement

print

System.out.println("Hello World");

class Main {

main() {

} }