

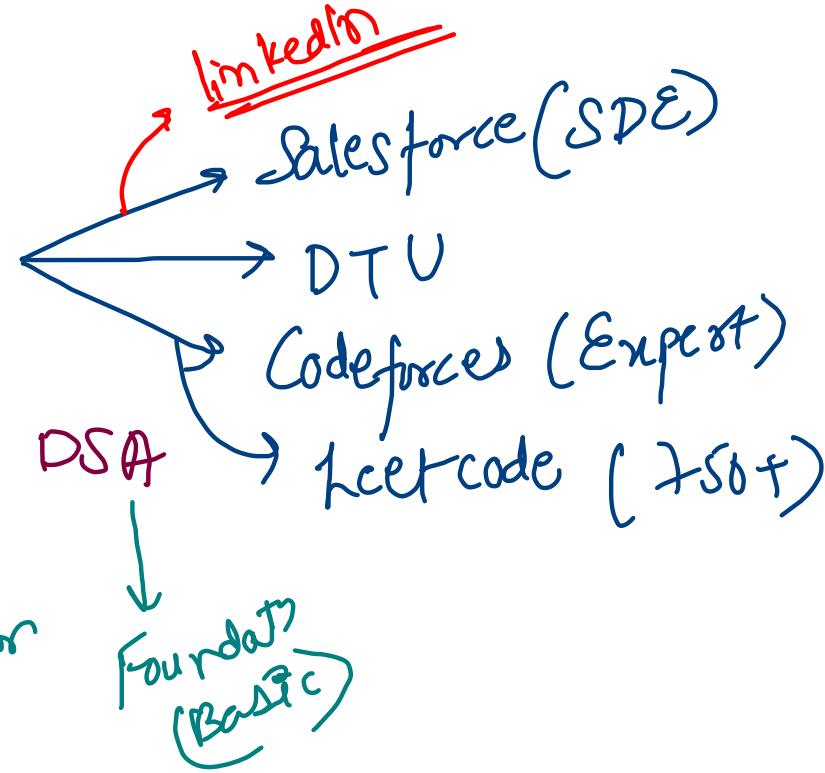
Introduction

Aashit Aggarwal

July 22 Java + DSA

Foundation
=

Foundation
(Basic)



5-15 VA

Syllabus (3 months)

→ Java Programming language

If-else
for loops
pattern

→ Abdul Basit
Java → Udemy
(300+ video)

→ Arrays (1D & 2D) , Strings & StringBuilder → Time & Space Complexity

→ Searching & Sorting → Two Pointers → PrefixSum

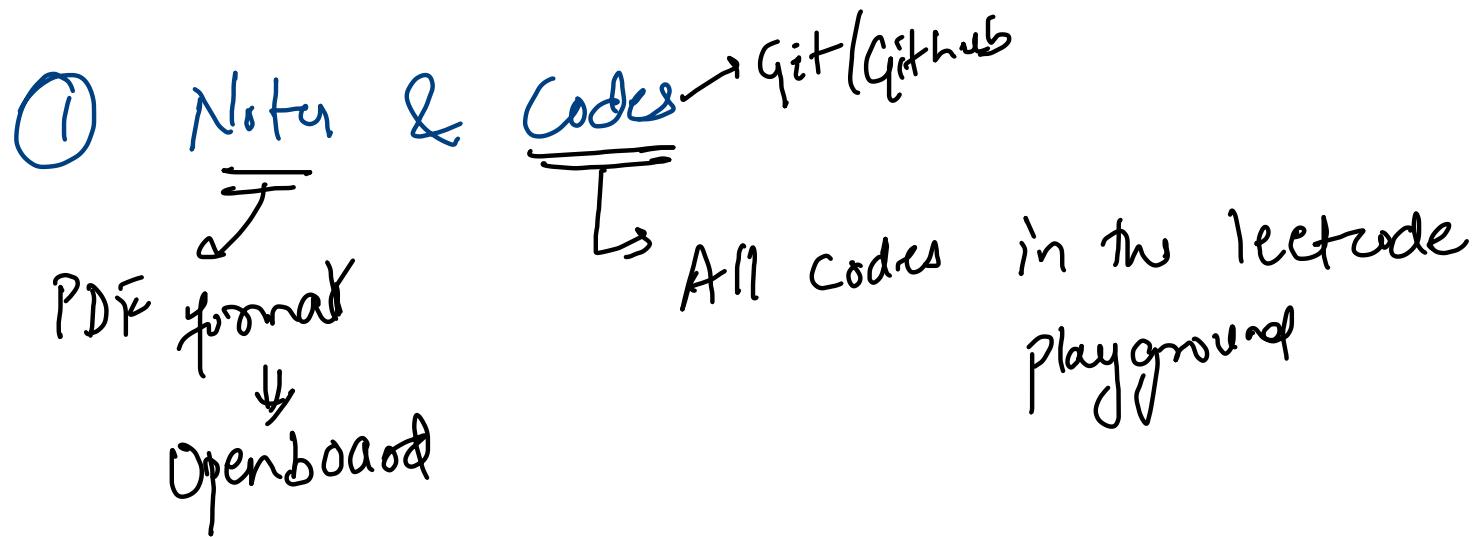
→ Object oriented programming in Java + Projects

→ Collection framework

→ Stack & Queue
linked list

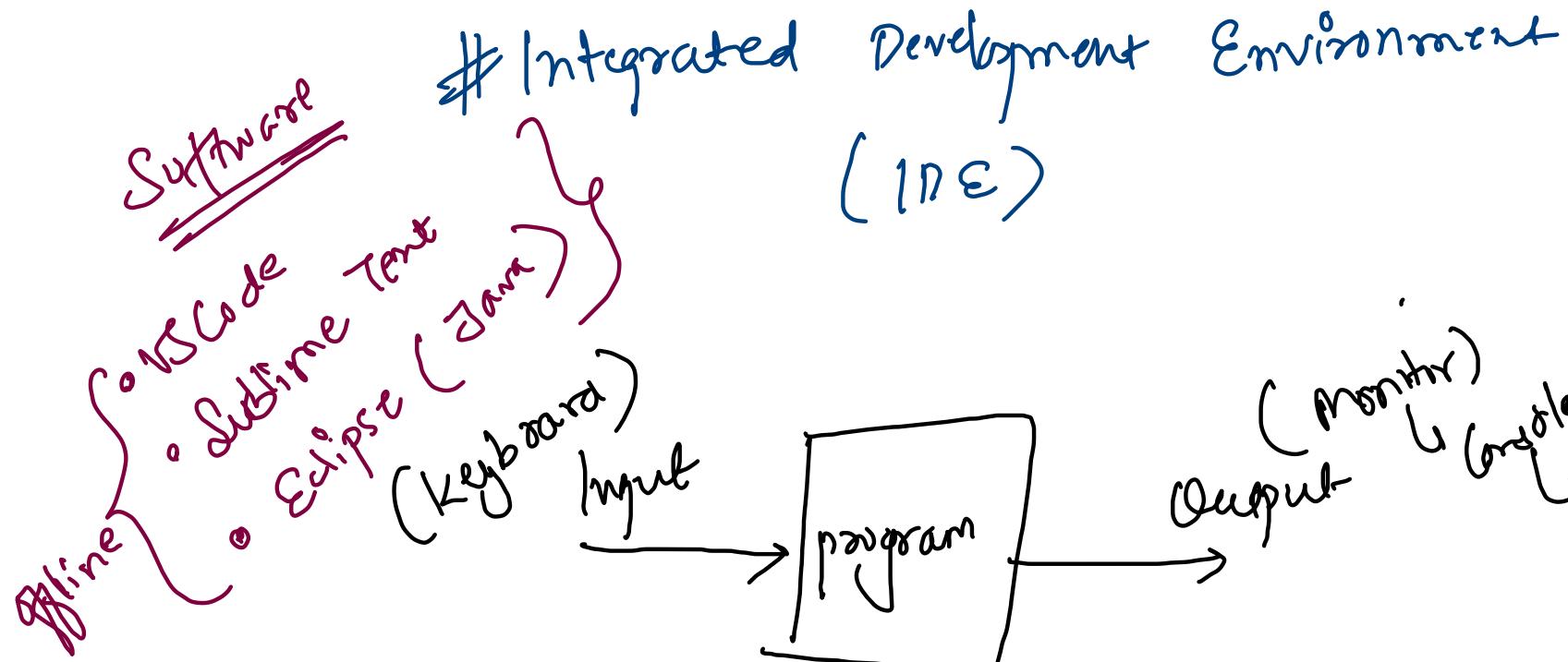
HashMap & Heap

→ Recursion &
Backtracking



② Recordings
→ Geelster Portal

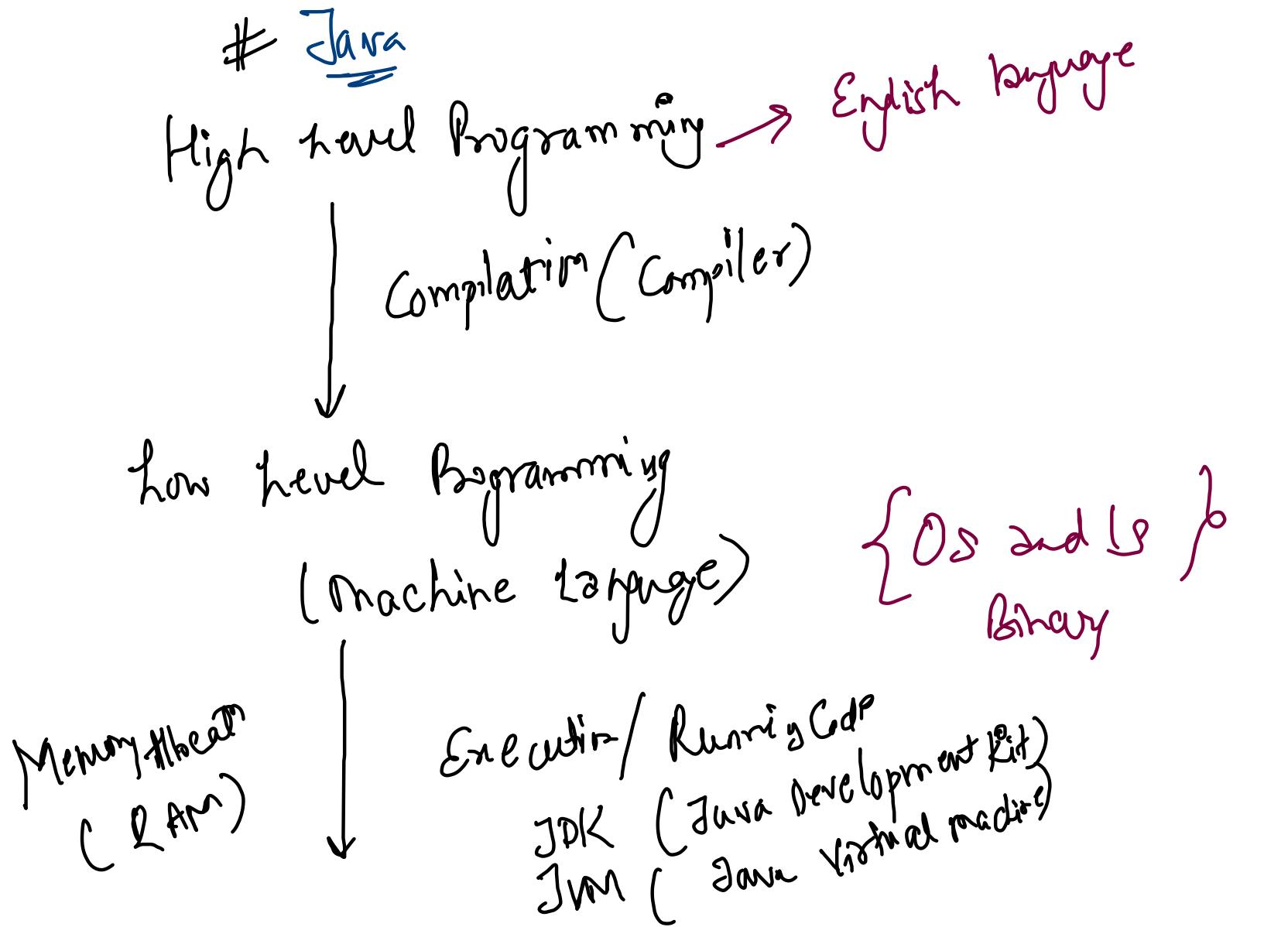
③ WhatsApp Group / Slack / Salesforce



Online { Leetcode
Geeks }

Programmers ✓
C Code ✓ & C

1.5 - 2 years forward
→ freelancing
→ Internship
→ GFG (Frederick City)
→ Repcoding
→Scaler
→ Unstoppable (DZone)
→ GeeksforGeeks (year)



10 + 15
Storage

Output

①

System.out.println ("Archit Agarwal");

↑
Capital

↳ Newline

②

System.out.print ("Geekster");

↳ Same line

Point 2

* * * * *
- - - *

- - *

- *

* * * * *

hw1

Point X

* *
* *
* *
* *

Bin H

* *
* *
* * * * *
* *

System.out.println("*****")
" " println("*")
" " println()
" " println()
" " println(>)

hw2

Live Teaching
20.1.30.
Chats
Unmute (raise Hand)

Teaching Assistant
↳ Shreya

- Doubts (7:30 - 8:00 PM) 1:1
- Live Doubts (Chat)
- + HackerRank Assignments & Tests

- Operations in Java*
- Addition Strings
Concatenation
- ① Arithmetic operators ($+$, $-$, $/$, $*$, $\%$)
 - ② Comparison operators ($<$, $>$, $<=$, $>=$, $==$, $!=$)
 - ③ Assignment operators ($=$)
 - ④ Logical operators (AND ($&&$), OR ($||$), NOT ($!$))

for the $\&\&$ result to
be true, both operands
should be
true

Truth table

AND ($\&\&$)

T	T $\&\&$ T	T $\&\&$ F
	= T	= F
F	F $\&\&$ T	F $\&\&$ F
	= F	= F

T F

Logical Operators

Truth table

OR ($\|$)

If any operand
is true, then
the result is
true.

Truth Table

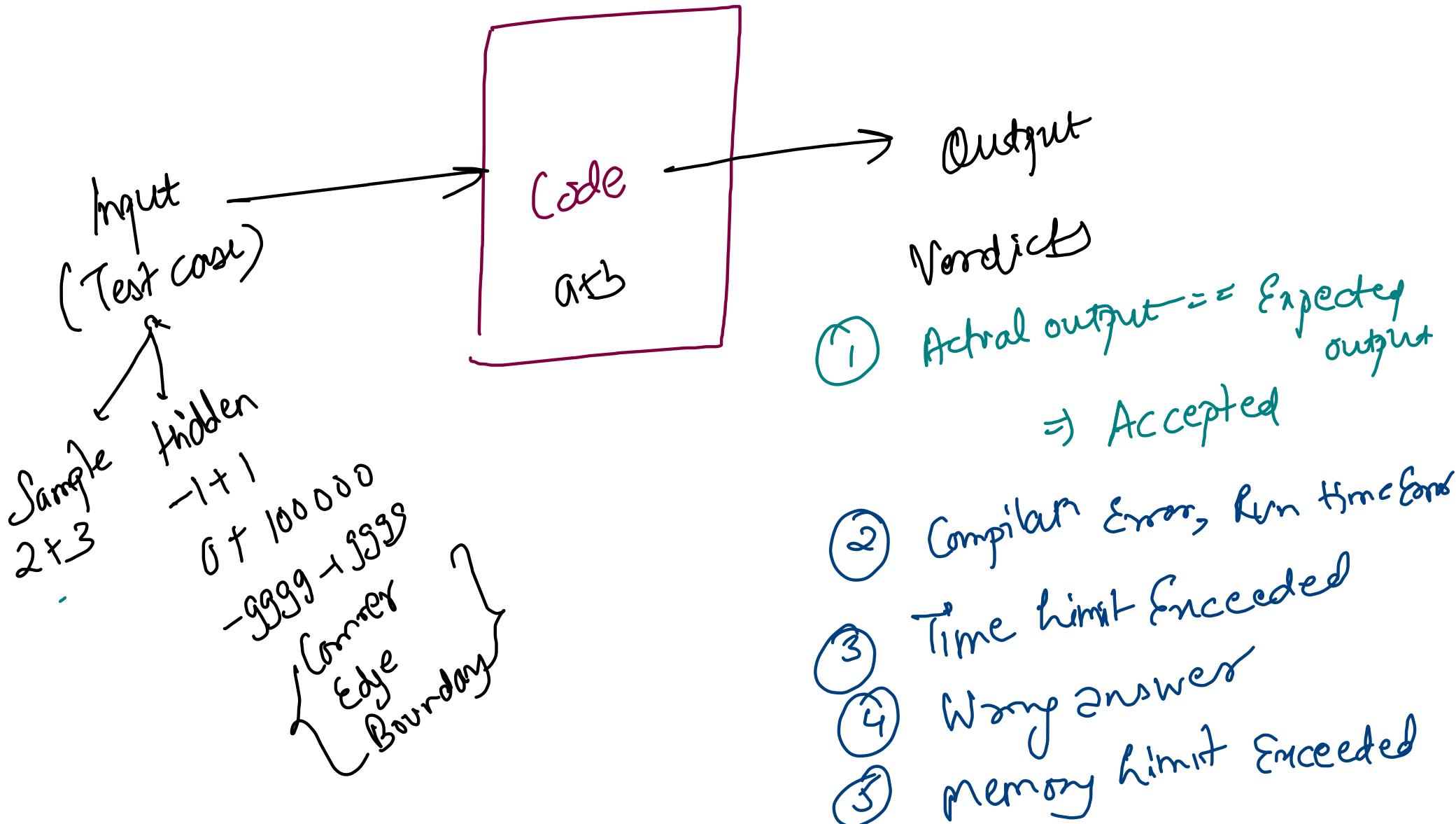
NOT (\sim)

T	T $\ $ T	T $\ $ F
	= T	= T
F	F $\ $ T	F $\ $ F
	= T	= F

T F

T	F
F	T

HackerRank (Online Judge)



Question

- Problem statement
 - Input format
 - Output format
 - Constraints ~~if~~
- Time limit Exceeded ↗
- Accepted ↗

↓ ↓
50 ,
↓ ↓
3.14 .
↓
Hello World

```
Scanner scn = new Scanner(System.in);
int a = scn.nextInt(); → 50
double b = scn.nextDouble(); → 3.14
scn.nextLine(); → "" (empty)
String c = scn.nextLine(); → "HelloWorld"
```

```
System.out.println("String: " + c);
System.out.println("Double: " + b);
System.out.println("Int: " + a);
```

~~Credits~~ Evaluation (Referrals)

- ① Attendance (CW Questions + presence in Zoom
+ feedback ~~due~~)
- ② Homework questions (HackerRank) \Rightarrow Assignment
- ③ Weekly Tests (HackerRank \Rightarrow Teaching Assistant) Sunday
- ④ Suggestions from Instructor / Teaching Assistant

Conditional statements

Integer Age

$\geq 18 \Rightarrow$ true "Adult"

$< 18 \Rightarrow$ false "Below Age"

System ($age \geq 18$)

Age

$50 > 40$

Salary

$50000 \geq 30000$

Rich & Adult

Age > 40 & salary $\geq 30k$

$50 > 40$

$20000 < 30000$

Adult

Age > 40 & salary $< 30k$

$30 \leq 40$

$50000 \geq 12000$

Rich & Young

Age ≤ 40 & salary $\geq 30k$

$30 \leq 40$

$10000 < 12000$

Young

Age ≤ 40 & salary $< 30k$

Follow up

Age

$50 > 40$

Salary

$\$0000 \geq 30000$

$50 > 40$

$20000 < 30000$

$30 \leq 40$

$\$0000 \geq 12000$

$30 \leq 40$

$10000 < 12000$

Gender

Male/female

Male/female

M/F

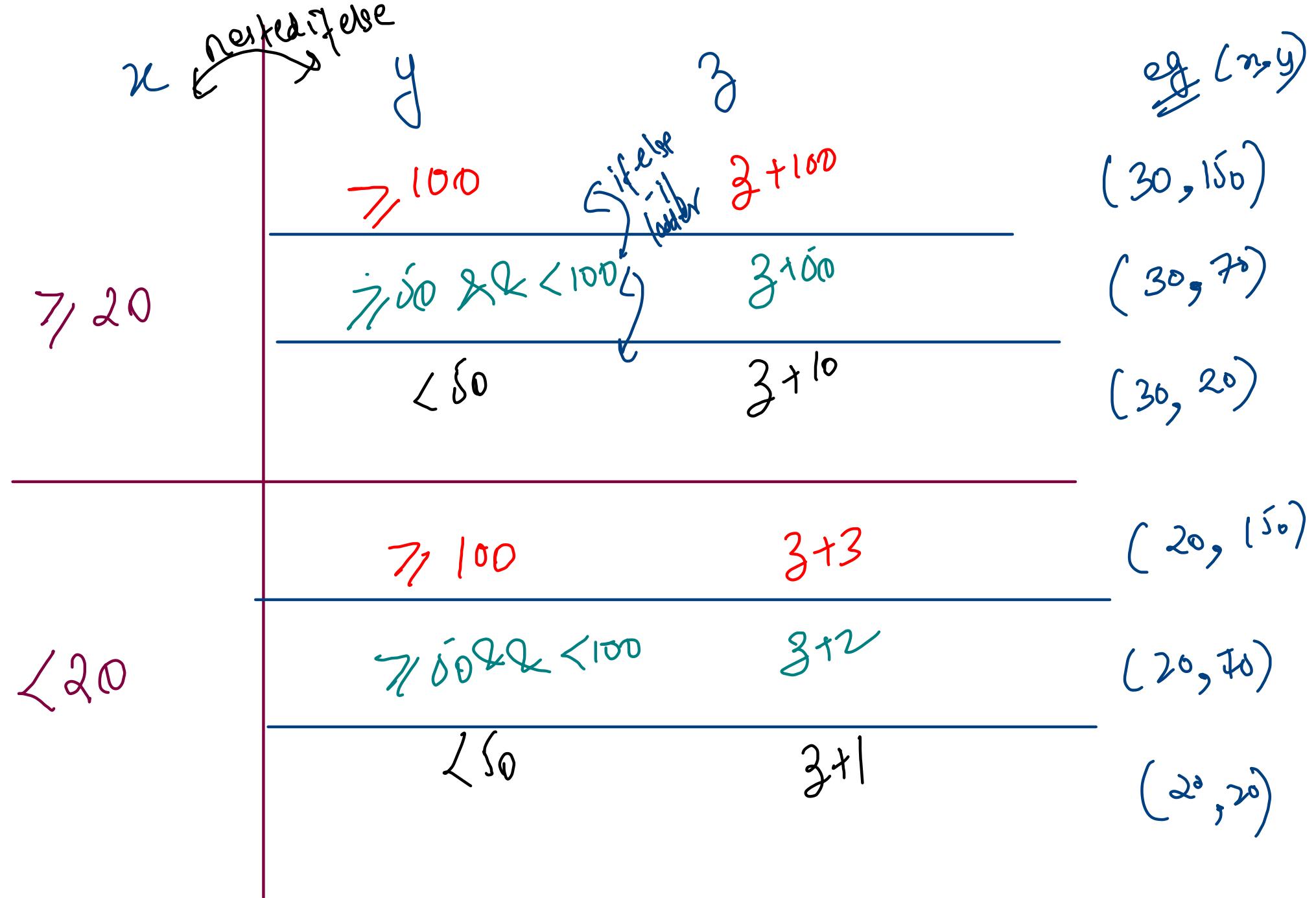
M/F

Rich & Adult

Adult

Rich & Young

Young



! @ # \$ % ^ & * ...
 Symbols

Special characters
 → tab, space,
 return,
 esc

emoji →


English Alphabets		
'a'	-	'z'
'A'	-	'Z'

~~'aa'~~ ~~'aa'~~
~~'aa'~~ ~~'aa'~~

Characters

Devnagri (Hindi)

अ - अ

ए - ए - ए - ...

digits

'0'

'1'

'2'

:

'g'

अ

ग

int marks = 95
 salary = 30000

Unicode → All characters
ASCII

Hindi, Telugu,
Emojis, ---

ASCII

American Standard Code for
Information Interchange

char → ASCII → 95 → 101111
int decimal o's l's binary

char → ASCII → 33 → 100011
int decimal

'0' → 0
int binary
 $\frac{1}{2} \pi r^2$ → 1
int binary
a → ASCII → 97 → 1100001
int decimal
0's l's binary

A → ASCII → 65 → ---
char int decimal
'0' → ASCII → 48 → 110000
char int binary

Dec	Char	Dec	Char	Dec	Char	Dec	Char
0	NUL (null)	32	SPACE	64	@	96	`
1	SOH (start of heading)	33	!	65	A	97	a
2	STX (start of text)	34	"	66	B	98	b
3	ETX (end of text)	35	#	67	C	99	c
4	EOT (end of transmission)	36	\$	68	D	100	d
5	ENQ (enquiry)	37	%	69	E	101	e
6	ACK (acknowledge)	38	&	70	F	102	f
7	BEL (bell)	39	'	71	G	103	g
8	BS (backspace)	40	(72	H	104	h
9	TAB (horizontal tab)	41)	73	I	105	i
10	LF (NL line feed, new line)	42	*	74	J	106	j
11	VT (vertical tab)	43	+	75	K	107	k
12	FF (NP form feed, new page)	44	,	76	L	108	l
13	CR (carriage return)	45	-	77	M	109	m
14	SO (shift out)	46	.	78	N	110	n
15	SI (shift in)	47	/	79	O	111	o
16	DLE (data link escape)	48	0	80	P	112	p
17	DC1 (device control 1)	49	1	81	Q	113	q
18	DC2 (device control 2)	50	2	82	R	114	r
19	DC3 (device control 3)	51	3	83	S	115	s
20	DC4 (device control 4)	52	4	84	T	116	t
21	NAK (negative acknowledge)	53	5	85	U	117	u
22	SYN (synchronous idle)	54	6	86	V	118	v
23	ETB (end of trans. block)	55	7	87	W	119	w
24	CAN (cancel)	56	8	88	X	120	x
25	EM (end of medium)	57	9	89	Y	121	y
26	SUB (substitute)	58	:	90	Z	122	z
27	ESC (escape)	59	;	91	[123	{
28	FS (file separator)	60	<	92	\	124	
29	GS (group separator)	61	=	93]	125	}
30	RS (record separator)	62	>	94	^	126	-
31	US (unit separator)	63	?	95	_	127	DEL

ASCII table

'A' - 'Z'

65 - 90

'a' - 'z'

97 - 122

'0' - '9'

48 - 57