

* logical operators : (AND, OR, NOT)

are both satisfying

A AND B

"Sarah has a driver's license
AND good vision"

A Possible values

B

AND	A	B	A AND B
A	TRUE	TRUE	T
TRUE	TRUE	F	F
FALSE	FALSE	FALSE	FALSE

Results of operation, depending on 2 variables

true when ALL are true

A ↑

A	B	A and B
T	T	T
T	F	F
F	T	F
F	F	F

EXAMPLE:

A: Sarah has a driver's license T
B: Sarah has good vision F

Boolean variables that can be either TRUE or FALSE

Can Sarah drive the car?

- only when she has a driver's license and also good vision

* And \Rightarrow True only when all the statements are true.

- Both statements must be true.

A OR B

"Sarah has a driver's license
OR good vision"

A

OR	TRUE	FALSE
TRUE	TRUE	TRUE
FALSE	TRUE	FALSE



true when **ONE** is true

either one of them
must satisfy



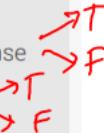
A	B	A OR B
T	T	T
T	F	T
F	T	T
F	F	F

* False only when all
statements are False.

EXAMPLE:

A: Sarah has a driver's license

B: Sarah has good vision



Boolean variables that can
be either TRUE or FALSE

(reverse)

NOT A, NOT B

$$T \rightarrow F$$

$$F \rightarrow T$$



Inverts true/false value

A	B	NOT A	NOT B
T	T	F	F
T	F	F	T
F	T	T	F
F	F	T	T

EXAMPLE :

BOOLEAN VARIABLES

👉 A: Age is greater or equal 20 → false

👉 B: Age is less than 30 → true

age = 16

$(16 \geq 20)$

$(16 < 30)$

- ① $\neg A \Rightarrow \neg \text{false} \Rightarrow \boxed{\text{true}}$
- ② $A \text{ and } B \Rightarrow \text{false and true} \Rightarrow \boxed{\text{false}}$
- ③ $A \text{ or } B \Rightarrow \text{false or true} \Rightarrow \boxed{\text{true}}$
- ④ $\neg A \text{ and } B \Rightarrow \neg \text{false and true}$
 $\Rightarrow \text{true and true} \Rightarrow \boxed{\text{true}}$
- ⑤ $A \text{ or } \neg B \Rightarrow \text{false or } \neg \text{true}$
 $\Rightarrow \text{false or false} \Rightarrow \boxed{\text{false}}$

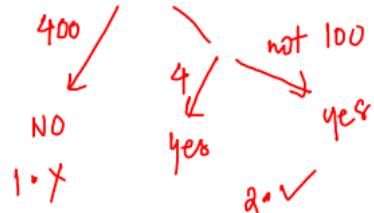
* Leap Year:

what? (366 days) (Feb 29th)

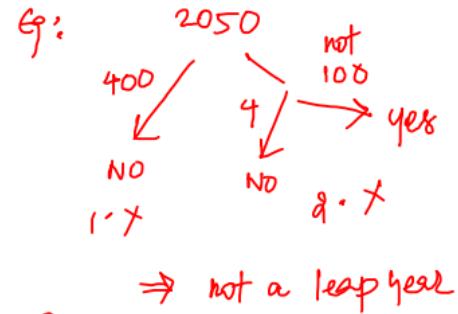
1. divisible by 400 (or) → A

2. divisible by 4 and not divisible by 100 → B
 ↓ C ↓ D

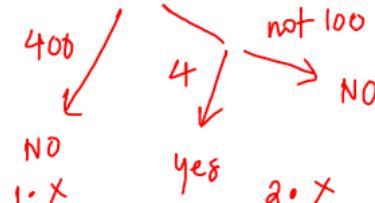
Eg: 2024



⇒ Leap Year



Eg: 2100



⇒ not a leap year

A or B

A or (C and D)

A ⇒ year % 400 == 0

C ⇒ year % 4 == 0

D ⇒ year % 100 != 0

A || (C & D)

* Which Case :

"A" → Capital → 1

"a" → LowerCase → 0

"#", "@" → not alphabet → -1
"↑" →

How do I know ch is Capital?

"A" - "z" \Rightarrow [65, 90]

⇒ If ch is a capital alphabet it should fit in the above range

LowerCase \Rightarrow [97, 122]

* Given a number check whether it is in the given range or not

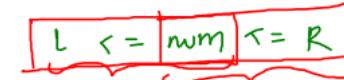
num = 12 → false
 \downarrow
= 612 → true

[525, 934]

[L, R]

Mathematically,

$$525 \leq num \leq 934$$



✓ L <= num
✗ num <= R

If (525 <= num & num <= 934) {

 cl("Yes");
}

else {

 cl("No");
}

* ASCII Values :

Const ch = "A";



Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr				
0	0 000	000	NUL (null)	32	20 040	0#32;	Space	64 40 100 0#64;	8	96 60 140 0#96;	;	140 0#96;	,	10	64 41 101 0#65;	A	97 61 141 0#97;	A				
1	1 001	001	SOH (start of heading)	33	21 041	0#33;	!	65 41 101 0#65;	A	97 61 141 0#97;	A	140 0#96;	,	11	65 42 102 0#66;	B	98 62 142 0#98;	B				
2	2 002	002	STX (start of text)	34	22 042	0#34;	"	66 42 102 0#66;	B	98 62 142 0#98;	B	140 0#96;	,	12	35 23 043 0#35;	#	67 43 103 0#67;	C	99 63 143 0#99;	C		
3	3 003	003	ETX (end of text)	35	23 043	0#35;	#	67 43 103 0#67;	C	99 63 143 0#99;	C	140 0#96;	,	13	36 24 044 0#36;	\$	68 44 104 0#68;	D	100 64 144 0#100;	D		
4	4 004	004	EOT (end of transmission)	36	24 044	0#36;	\$	68 44 104 0#68;	D	100 64 144 0#100;	D	140 0#96;	,	14	5 005 ENQ (enquiry)	?	69 45 105 0#69;	E	101 65 145 0#101;	E		
6	6 006	006	ACK (acknowledge)	37	25 045	0#37;	?	69 45 105 0#69;	E	101 65 145 0#101;	E	140 0#96;	,	15	7 007 BEL (bell)	;	70 46 106 0#70;	F	102 66 146 0#102;	F		
8	8 010	010	BS (backspace)	39	27 047	0#39;	'	71 47 107 0#71;	G	103 67 147 0#103;	G	140 0#96;	,	9	9 011 TAB (horizontal tab)	{	72 48 110 0#72;	H	104 68 150 0#104;	H		
10	A 012 LF (NL line feed, new line)	012		40	28 050	0#40;	{	73 49 111 0#73;	I	105 69 151 0#105;	I	140 0#96;	,	11	41 29 051 0#41;	}	74 4A 112 0#74;	J	106 6A 152 0#106;	J		
11	B 013 VT (vertical tab)	013		42	2A 052	0#42;	*	74 4A 112 0#74;	J	106 6A 152 0#106;	J	140 0#96;	,	12	C 014 FF (NP form feed, new page)	+	75 4B 053 0#43;	K	107 6B 153 0#107;	K		
13	D 015 CR (carriage return)	014		44	2C 054	0#44;	/	75 4B 113 0#75;	K	107 6B 153 0#107;	K	140 0#96;	,	14	E 016 SO (shift out)	,	76 4C 055 0#45;	-	76 4C 114 0#76;	L	108 6C 154 0#108;	L
15	F 017 SI (shift in)	015		45	2D 055	0#45;	-	77 4D 115 0#77;	M	109 6D 155 0#109;	M	140 0#96;	,	16	10 020 DLE (data link escape)	:	78 4E 056 0#46;	.	78 4E 116 0#78;	N	110 6E 156 0#110;	N
17	11 021 DC1 (device control 1)	020		46	2E 057	0#47;	/	79 4F 117 0#79;	O	111 6F 157 0#111;	O	140 0#96;	,	18	12 022 DC2 (device control 2)	;	80 50 120 0#80;	P	112 70 160 0#112;	P		
19	13 023 DC3 (device control 3)	021		47	2F 057	0#47;	/	81 51 121 0#81;	Q	113 71 161 0#113;	Q	140 0#96;	,	20	20 14 024 DC4 (device control 4)	0	80 50 120 0#80;	P	112 70 160 0#112;	P		
21	21 15 025 NAK (negative acknowledge)	022		48	30 060	0#48;	0	82 52 122 0#82;	R	114 72 162 0#114;	R	140 0#96;	,	22	22 16 026 SYN (synchronous idle)	1	83 53 123 0#83;	S	115 73 163 0#115;	S		
23	23 17 027 ETB (end of trans. block)	023		49	31 061	0#49;	1	84 54 124 0#84;	T	116 74 164 0#116;	T	140 0#96;	,	24	24 18 030 CAN (cancel)	2	85 55 125 0#85;	U	117 75 165 0#117;	U		
25	25 19 031 EM (end of medium)	024		50	32 062	0#50;	2	86 56 126 0#86;	V	118 76 166 0#118;	V	140 0#96;	,	26	26 1A 032 SUB (substitute)	3	87 57 127 0#87;	W	119 77 167 0#119;	W		
27	27 1B 033 ESC (escape)	025		51	33 063	0#51;	3	88 58 130 0#88;	X	120 78 170 0#120;	X	140 0#96;	,	28	28 1C 034 FS (file separator)	4	89 59 131 0#89;	Y	121 79 171 0#121;	Y		
29	29 1D 035 GS (group separator)	026		52	34 064	0#52;	4	90 60 132 0#90;	Z	122 7A 172 0#122;	Z	140 0#96;	,	30	30 1E 036 RS (record separator)	5	91 5B 133 0#91;	[123 7B 173 0#123;	[
31	31 1F 037 US (unit separator)	027		53	35 065	0#53;	5	92 5C 134 0#92;	\	124 7C 174 0#124;	\	140 0#96;	,	32	54 36 066 0#54;	6	93 5D 135 0#93;]	125 7D 175 0#125;]		
				55	37 067	0#55;	7	94 5E 136 0#94;	^	126 7E 176 0#126;	~		,	56	38 070 0#56;	8	95 5F 137 0#95;	_	127 7F 177 0#127;	DEL		

Const S = "ACBD"



"A" - "Z" "a" - "z"

[65, 90] [97 - 122]
↑25 ↑25

→ String • fromCharCode(ascii)
(ASCII → character)

→ "A" = charCodeAt(0)
(ch → ASCII)

* Big Light:

Gian grows by 3m in every second

①	G: 5m	s: 7m
	$v_1: 3 \text{ m/s}$	$v_2: 2 \text{ m/s}$

time, $t = 0$ $\begin{matrix} 5\text{m}, 7\text{m} \\ +3 \end{matrix} \rightarrow 2\text{m}$

$t = 1$ $\begin{matrix} 8\text{m}, 9\text{m} \\ +3 \end{matrix} \rightarrow 1\text{m}$

$t = 2$ $\begin{matrix} 11\text{m}, 11\text{m} \\ +3 \end{matrix} \rightarrow 0\text{m}$

\Rightarrow op: true (they meet/Equal)
at $t = 2$

$$\star t = \frac{h_2 - h_1}{v_1 - v_2} = \frac{7 - 5}{3 - 2}$$

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

② G: 5m, s: 7m

$v_1: 2 \text{ m/s}$, $v_2: 3 \text{ m/s}$

time, $t = 0$ $\begin{matrix} 5\text{m}, 7\text{m} \\ +3 \end{matrix} \rightarrow 2\text{m}$

$t = 1$ $\begin{matrix} 7\text{m}, 10\text{m} \\ +3 \end{matrix} \rightarrow 9\text{m}$

$t = 2$ $\begin{matrix} 9\text{m}, 13\text{m} \\ +3 \end{matrix} \rightarrow 4\text{m}$

$t = 3$ $\begin{matrix} 11\text{m}, 16\text{m} \\ +3 \end{matrix} \rightarrow 5\text{m}$

$t = 4$ $\begin{matrix} 13\text{m}, 19\text{m} \\ +3 \end{matrix} \rightarrow 6\text{m}$

$\vdots \quad \vdots \quad \vdots$

op: false

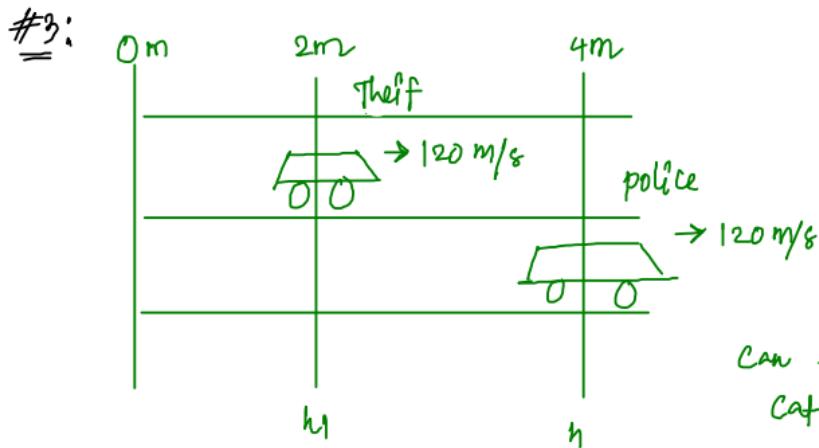
$$h_1, h_2$$

$$v_1, v_2$$

① $t = 0, h_1 = h_2 \rightarrow \text{true}$

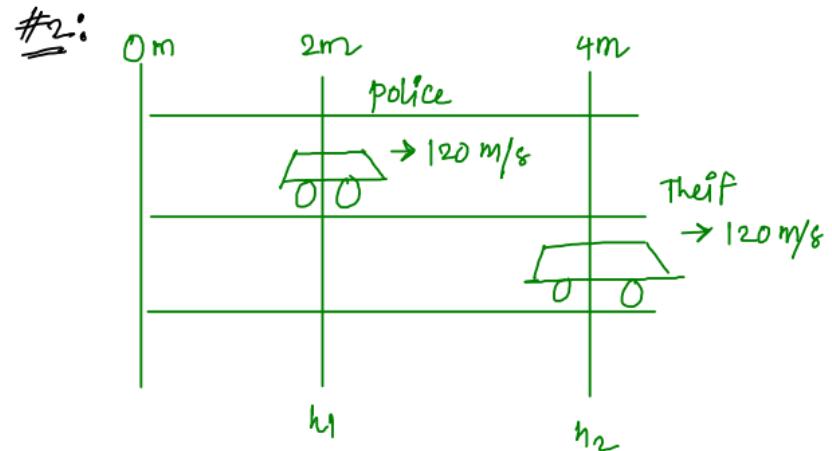
② $h_1 < h_2, v_1 < v_2 \rightarrow \text{false}$

③ $h_2 < h_1, v_2 > v_1 \rightarrow \text{false}$



Can the police
catch thief

\Rightarrow Cannot Catch



Can the police
catch thief

\Rightarrow Theif is already
a head of police
and he is moving
faster than police

↓
Cannot catch

$$\textcircled{4} \quad h_1 < h_2, v_1 > v_2$$

(a)

\rightarrow ~~the~~

$$h_2 < h_1, v_2 > v_1$$

* We cannot predict they will meet at same pos or not.
 \Rightarrow maths

$$h_1 = 2\text{m}$$

$$h_2 = 3\text{m}$$

$$v_1 = 7\text{m/s}$$

$$v_2 = 2\text{m/s}$$

$$t = 0, 2\text{m}, 3\text{m} \rightarrow -1\text{m}$$

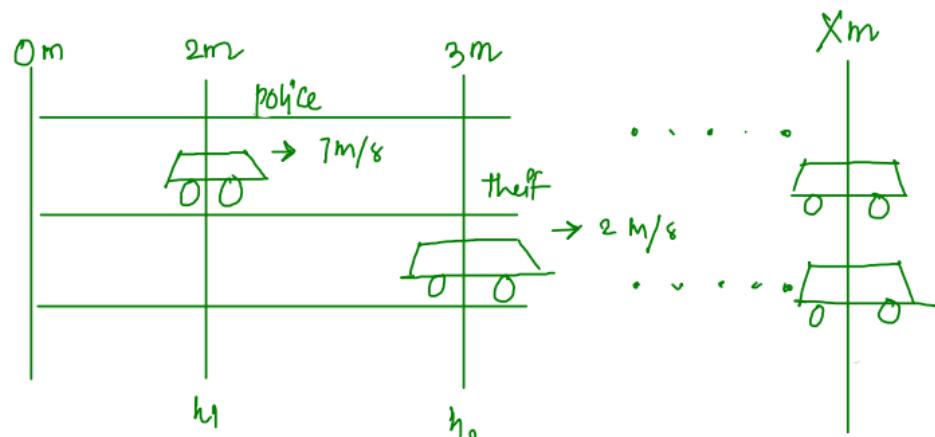
$$t = 1, 9\text{m}, 5\text{m} \rightarrow 4\text{m}$$

$$t = 2, 16\text{m}, 7\text{m} \rightarrow 9\text{m}$$

$$t = 3, 23\text{m}, 9\text{m} \rightarrow 14\text{m}$$

Op: false

$$t = \frac{h_2 - h_1}{v_1 - v_2} = \frac{3 - 2}{7 - 2} = \frac{1}{5} = 0.2\text{s}$$



* We can say that police will catch the thief but we are not sure abt equal positions.

Let us assume they will meet at some positions

After some 't' seconds,

$$\text{for } 1s, h_1 + v_1 = 5m + 2 = 7m$$

$$h_2 + v_2 = 3m + 3 = 6m$$

$$\text{for } 2s, h_1 + v_1 + v_1 = 5 + 2 + 2 = 9m$$

$$h_2 + v_2 + v_2 = 3 + 3 + 3 = 9m$$

$$\text{for } 3s, h_1 + v_1 + v_1 + v_1 = 5 + 2 + 2 + 2 = 11m$$

$$h_2 + v_2 + v_2 + v_2 = 3 + 3 + 3 + 3 = 12m$$

$$\text{By for } t's, h_1 + tv_1 = h_1^{\text{new}}$$

$$h_2 + tv_2 = h_2^{\text{new}}$$

\rightarrow When is $t \rightarrow$ Number

$$\text{only when } (h_2 - h_1) \% (v_1 - v_2) == 0$$

$$h_1 = 5m \quad h_2 = 3m$$

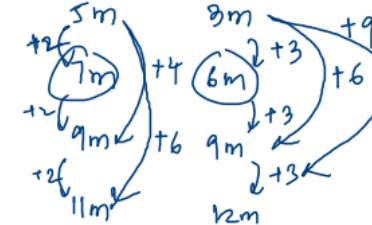
$$v_1 = 2m/s \quad v_2 = 3m/s$$

$$t=0$$

$$t=1$$

$$t=2$$

$$t=3$$



We assumed that they will meet after t seconds,

$$h_1^{\text{new}} = h_2^{\text{new}}$$

$$h_1 + tv_1 = h_2 + tv_2$$

$$tv_1 - tv_2 = h_2 - h_1$$

$$t(v_1 - v_2) = h_2 - h_1$$

$$t = \left(\frac{h_2 - h_1}{v_1 - v_2} \right)$$

Number \times (div) \uparrow
Fraction \times (not div) \downarrow

* How to Approach ?

1. Understand Qs clearly

→ Examples (ip, op)

→ Constraints (TestCases are limited to this)

2. * put your thoughts on paper

→ take some examples

→ analyse and obtain some insights

→ Forget coding think how a general person solves

3. Write some steps / have that logic in your mind

→ verify that with some examples (try to take examples where you feel that your logic fails).

4. Code → errors / failed testCases

* New problem

you will be able

to solve if you have
done something similar
earlier.

* how to build logic

↓
" practice different
Qs"

→ attend class

→ solve class problems
without seeing

↓
By understanding

→ Solve Assignment
Qs