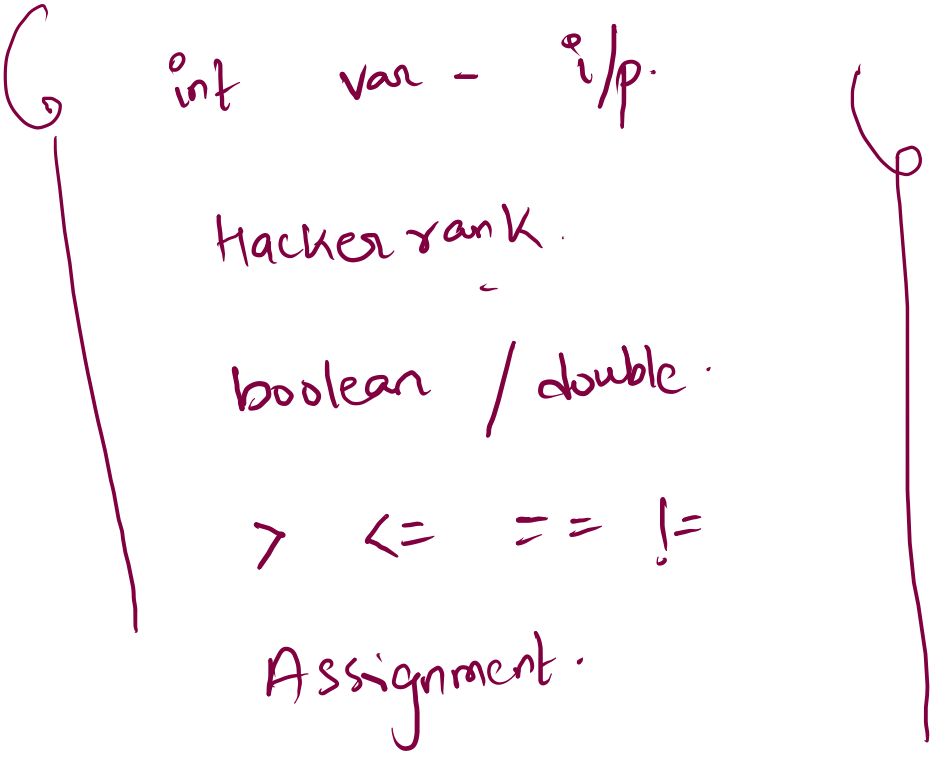
Yest.



Hackerrank.

4-5 today.

or && || (Truth)

maths

$2(x+y)$

↓

$2 * (x+y)$ ✓

xy

↓

$x * y$

Area & Perimeter of Rectangle.

Area → $l * b$

Perimeter → $2 * (l + b)$ ✓

Steps:

1. i/p → l & b. ✓
   ↳ scn.
2. area
3. peri
4. print both.

$2(l+b)$

con

✓ please follow

```java
import java.io.*;
import java.util.*;

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);

        int l = scn.nextInt();
        int b = scn.nextInt();

        int area = l * b;
        int perimeter = 2*(l + b);

        System.out.println(area);
        System.out.println(perimeter);

    }
}
```

F to C°

$$C^\circ \times \frac{9}{5} + 32 = F$$

Steps
1. → i/p
2. Convert
3. print.

4 F ⟶ C ?°

8:49 pm

~~8:54 pm~~

nextDouble();

$$C^\circ = (F - 32) \times \frac{5}{9}$$

Language: Java 8

```java
import java.io.*;
import java.util.*;

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);

        double far = scn.nextDouble();
        double cel = (far-32) * 5.0/9.0;

        System.out.println(cel);


    }
}
```

Add Last Digit

Sample Input 0

```
23456
9873
```

Sample Output 0

```
9
```

a = 23456 ⇒ 6

b = 9873 ⇒ +3

9

Steps:

1. i/p
2. (ld of both) add
3. print

13 % 3 = ? 1

a = 23456

```
3 | 13 | 4
    12
    ---
     1
```

```
10 | 23456 | 2345
     20
     ---
      34
      30
      ---
       45
       40
       --
        56
        50
        --
         6
```

last

n % 10

```java
import java.io.*;
import java.util.*;

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);

        int n = scn.nextInt();
        int m = scn.nextInt();

        int ans = (n % 10) + (m % 10);

        System.out.println(ans);

    }
}
```

Comments

→ Greater than 100 or not?

$n = 37$.

$n > 100 \rightarrow$ true

$37 > 100$

$n == 100$
$n < 100$] false.

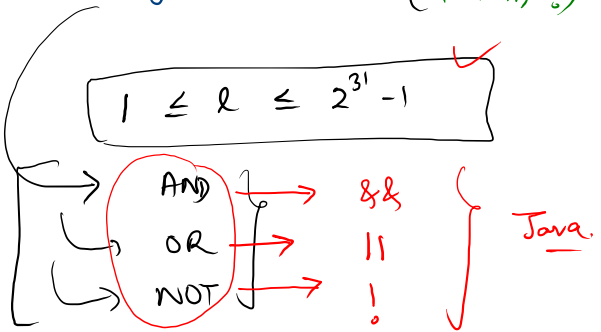
{ 1. i/p.
2. compare & print. }.

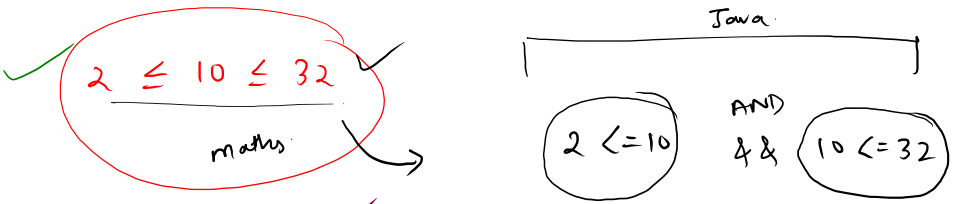
another.

Operators.

(Logical operators) (&&, ||, !)

Ar +
Com ==
>
<





&& → AND ⇒ (if both are true (1) then only ans. is true)

Truth Table

A	B	A && B
0	0	0
→ 0	1	0
→ 1	0	0
→ 1	1	1

0 → false
1 → true.

1 → true.

⇔

A	B	A && B
false	false	false.
false	true	false
true	false	false
true	true	true.

OR → ||

(if anyone of them are true then answer should be true).

A	B	A || B
0	0	0
0	1	1
1	0	1
1	1	1

<=>

A	B	A || B
false	false	false
false	true	true
true	false	true
true	true	true.

NOT (!) ⇒ (gives value if true ans is false & vice-versa).

A	!A
true	false
false	true

Key Points

AND → &&

:→ both true → ans. true.

OR → ||

↳ anyone true → ans true.

NOT - !

↳ opposite of value.

```java
boolean ans = (!(30==30) && (20 > 30)) || (20==20)   ;
```