We'll start by 9:00 pm.

L6 - Java
Loops Extra Class

RECAP

1. Given a value of R & K, print all the numbers from 1 to R that have sum of digits equal to K

2. While, do-while Loop

3. Given *target_cnt* and *target_sum*, print the first *target_cnt* numbers that have digit sum = *target_sum*

4. Check if a given number if prime or not?

Did you folks do homework?

Watched recordings? (for those who missed live)

Submitting on LeetCode etc.

given a number,
check if it is prime
or not?

Practice Continued

2153 ⇒ Sum of digits = 11

Sum
from 1 to N

N = 15 ⇒ 2, 3, 5, 7, 11, 13

Print all prime numbers from 1 to N.

```
for (int num = 2; num <= N; ++num) {
    // Check if num is prime
    boolean isPrime = true;
    for (int i = 2; i < num; ++i)
        if (num % i == 0)
            { isPrime = false; break; }
    // If yes, then print it
    if (isPrime) System.out.println(num);
}
```

N = 1 ⇒ Yes (2^0)

N = 6 ⇒ No

N = 8 ⇒ Yes

(2^3)

Given a number, check if it's a power of 2 or not

8 & 4

1, 2, 4, 8, 16, 32, 64, 128 . . . .

$2^5 = 32$

16

8

4

2

1

$2^9 \Rightarrow 512$

256

128

64

32

16

8

4

2

1

N_2 15 $\Rightarrow$ No

N_2 24 $\Rightarrow$ 12 $\Rightarrow$ 6 $\Rightarrow$ 3 $\Rightarrow$ No

N_2 $2^k * 3 * 7^2$ $\Rightarrow$ $3 * 7^2 \Rightarrow 147$

Let's say, the number is 2^K

N_2 $\underbrace{2 * 2 * 2 * 2 \cdots\cdots 2}_{K \text{ times}}$

$\frac{2^K}{2^1} \Rightarrow \frac{2^{K-1}}{2^1} \Rightarrow 2^{K-2} \Rightarrow 2^{K-3} \cdots\cdots 2^3, 2^2, 2^1, 1$

$2^0 \Rightarrow 1$

$2^1 \Rightarrow 2$

4

8

16

,

,

-8

$(-2)^3 \Rightarrow -8$

$2^3 \Rightarrow 8$

$2^{-3} \Rightarrow \frac{1}{8}$

gcd ⇒ greatest common divisor

⇑ ↓ +ve

Find HCF of 2 given numbers

↳ Highest common factor

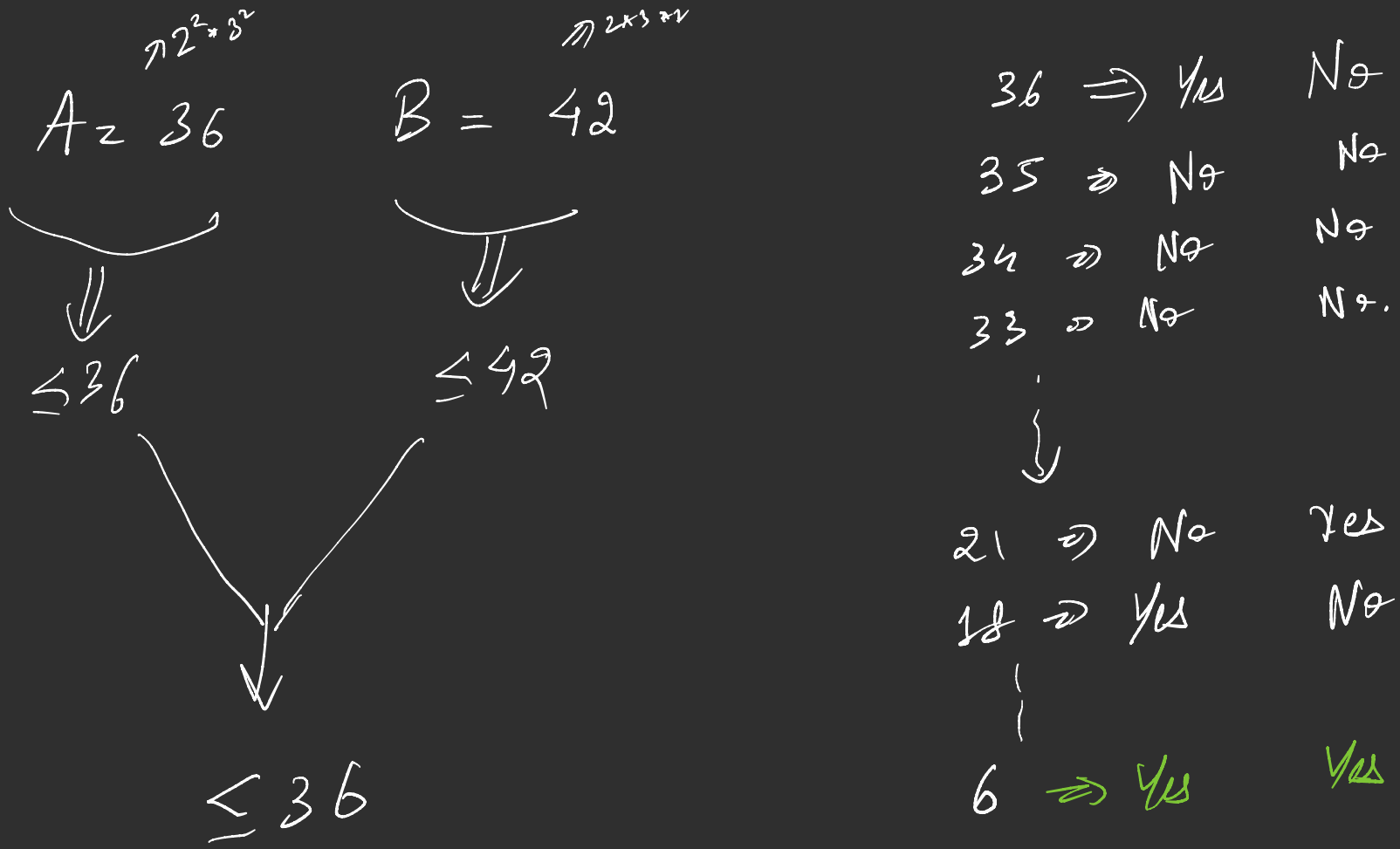
$A = 24, \quad B = 18$

$A = 18, \quad B = 36$

$\Rightarrow$ ans 18

1, 2, 3, 6

24


A = 36
B = 42
≤ 36
≤ 42
≤ 36
36 ⇒ Yes
No
35 ⇒ No
No
34 ⇒ No
No
33 ⇒ No
No.
21 ⇒ No
Yes
18 ⇒ Yes
No
6 ⇒ Yes
Yes