Time Complexity

21:29

Relation between Input Size & Running Time (Operations)

Scanner → input a variable "n"

```
for( int i=0 to n)
{
    print("hello"); → (1)
}
```

} 1×n times

1 Time Complexity

```
public static void main(String args[]) {
    Scanner sc = new Scanner(System.in);
    int n = sc.nextInt();

    for(int i=0; i<n; i++) {
        System.out.println("hello");
    }
}
```

input n

n times

time complexity

2 Time Complexity

BEST CASE

AVERAGE CASE

WORST CASE

2 Time Complexity

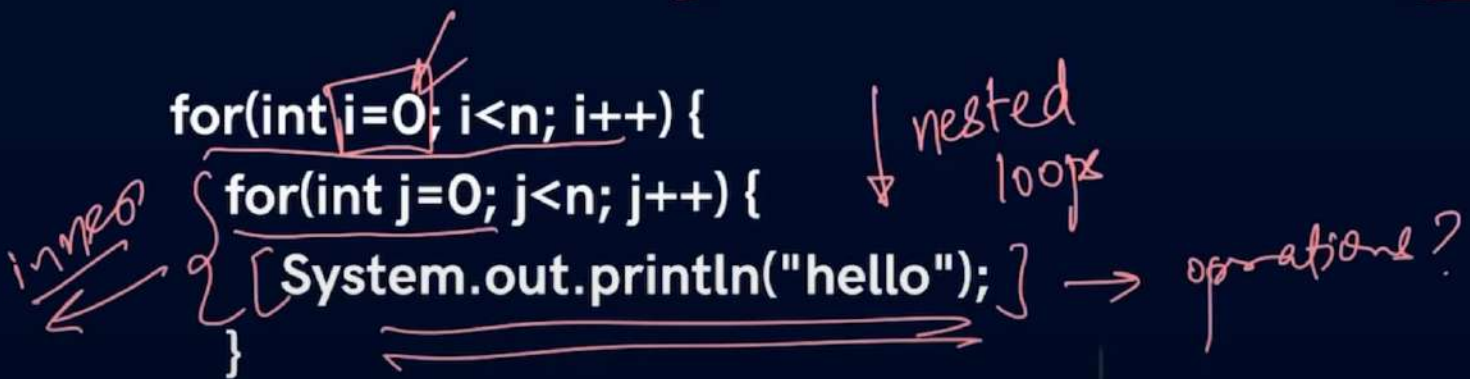
BEST CASE → $\Omega(1)$

AVERAGE CASE → $\Theta\left(\frac{n+1}{2}\right)$

WORST CASE → $O(n)$

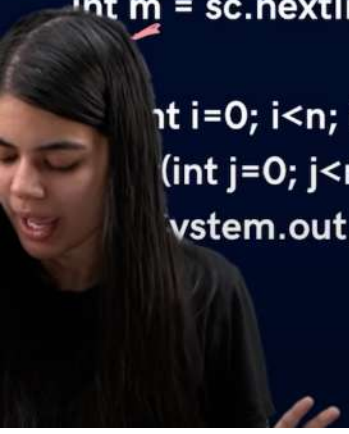
3 Time Complexity

```
public static void main(String args[]) {
    Scanner sc = new Scanner(System.in);
    int n = sc.nextInt();

    for(int i=0; i<n; i++) {
        for(int j=0; j<n; j++) {
            System.out.println("hello");
        }
    }
}
```



4 Time Complexity

```
public static void main(String args[]) {
    Scanner sc = new Scanner(System.in);
    int n = sc.nextInt();
    int m = sc.nextInt();

    nt i=0; i<n; i++) {
    (int j=0; j<m; j++) {
    ystem.out.println("hello");
```



i=1 j=0,1,2 --- m

i=3 j=0 to m

i=n j=0 --- m

[n times x m times]

n x m

time complexity O(n x m)

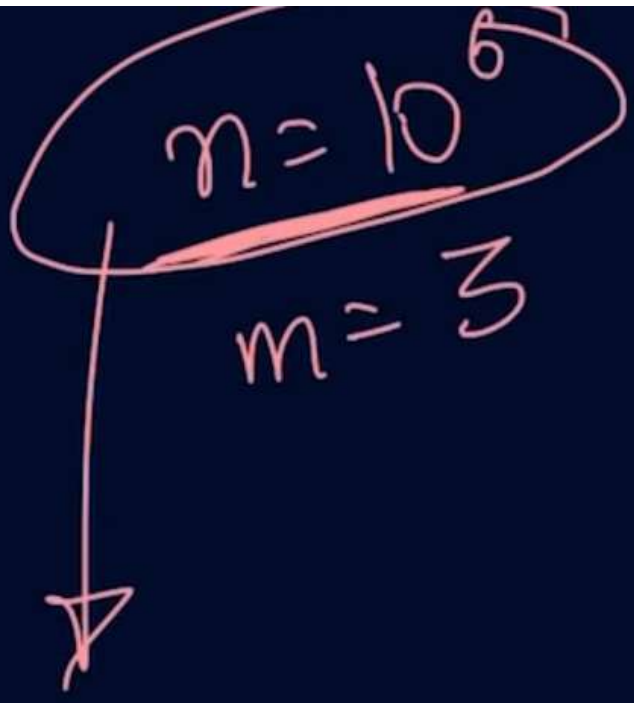
6 Time Complexity

```
public static void main(String args[]) {
    Scanner sc = new Scanner(System.in);
    int n = sc.nextInt();
    int m = sc.nextInt();

    for(int i=0; i<n; i++) {
        System.out.println("hello");
    }
    for(int j=0; j<m; j++) {
        System.out.println("hello");
    }
}
```

$O(n+m)$

$k \quad k$

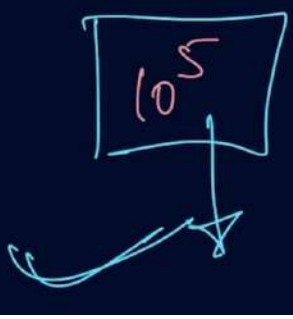
$2 \quad 3$



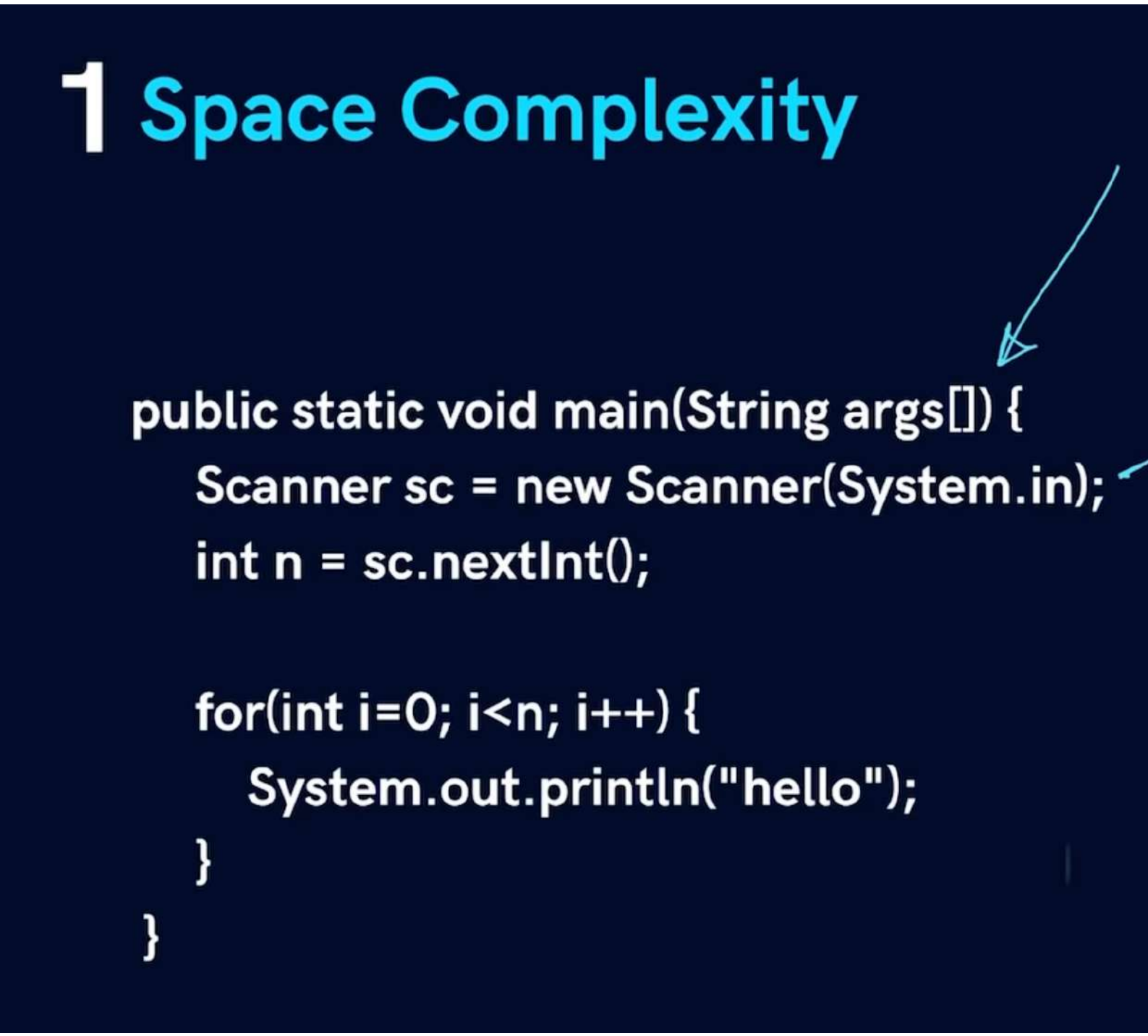
$O(n)$

7 Time Complexity

Compare :	O(n)	O(n^2)	O(n^3)
n = 1	1	1	1
n = 2	2	4	8
n = 3	3	9	27
10^5	10^5	10^{10}	10^{30}



1 Space Complexity



```
public static void main(String args[]) {
    Scanner sc = new Scanner(System.in);
    int n = sc.nextInt();

    for(int i=0; i<n; i++) {
        System.out.println("hello");
    }
}
```