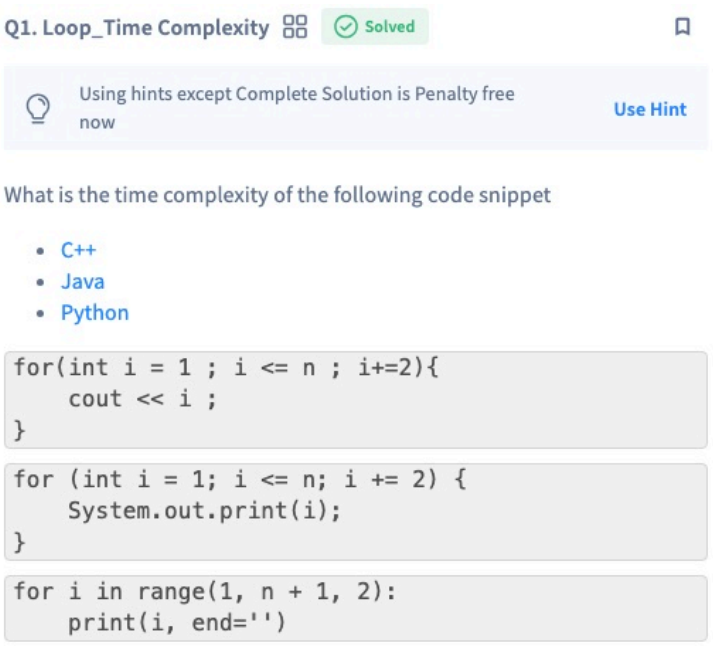
Time Complexity

(Introduction to Problem Solving I)

Assignments



Q1. Loop_Time Complexity Solved

Using hints except Complete Solution is Penalty free now

Use Hint

What is the time complexity of the following code snippet

- C++
- Java
- Python

```
for(int i = 1 ; i <= n ; i+=2){
    cout << i ;
}
```

```
for (int i = 1; i <= n; i += 2) {
    System.out.print(i);
}
```

```
for i in range(1, n + 1, 2):
    print(i, end='')
```

$$\Rightarrow \quad i \rightarrow \underbrace{1 \rightarrow 3 \rightarrow 5 \rightarrow 7 \dots \rightarrow}_{n \text{ times } \rightarrow \ n \text{ iterations}} N \longrightarrow \text{A.P with}$$

$C.D = 2$

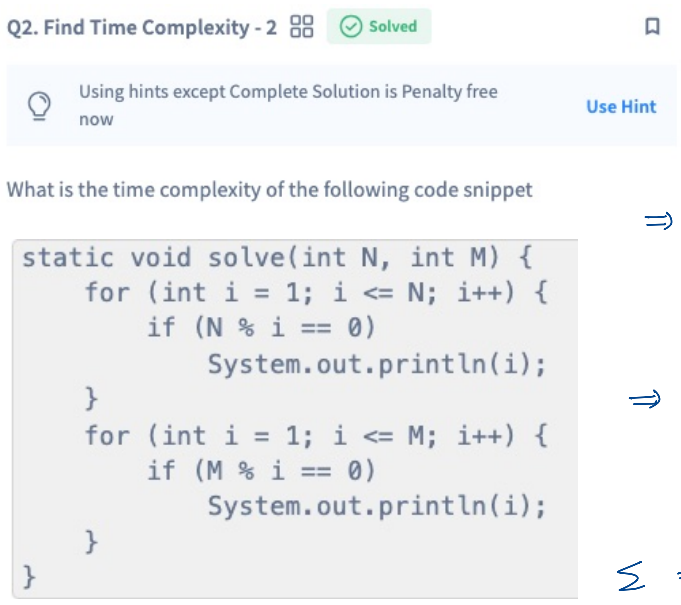
$a = 1$

$l = N$

$$l = a + (n-1)d \Rightarrow N = 1 + (n-1)2$$

$$\Rightarrow N = 1 + 2n - 2 \Rightarrow 2n = N + 1 \Rightarrow \boxed{n = \frac{N+1}{2}}$$

$$\Rightarrow O(N)$$



Q2. Find Time Complexity - 2

Solved

Using hints except Complete Solution is Penalty free now

Use Hint

What is the time complexity of the following code snippet

```
static void solve(int N, int M) {
    for (int i = 1; i <= N; i++) {
        if (N % i == 0)
            System.out.println(i);
    }
    for (int i = 1; i <= M; i++) {
        if (M % i == 0)
            System.out.println(i);
    }
}
```

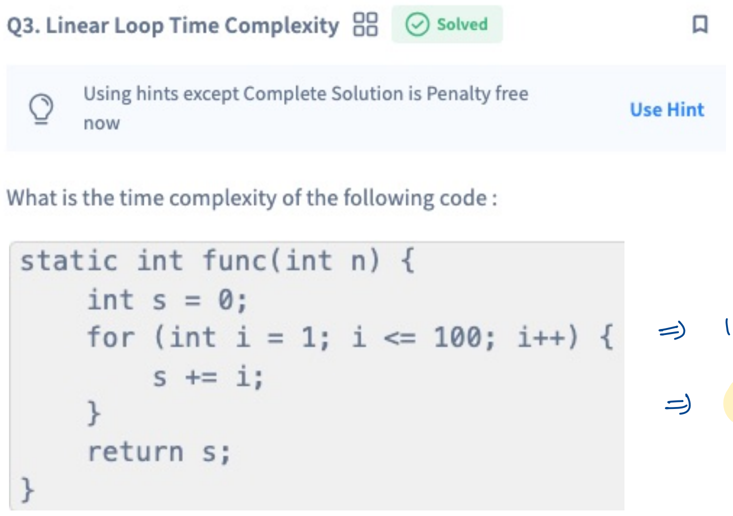
$\Rightarrow$ First Loop

N iterations

$\Rightarrow$ Second Loop

M iterations

$\Sigma = N+M = O(N+M)$



Q3. Linear Loop Time Complexity

Solved

Using hints except Complete Solution is Penalty free now

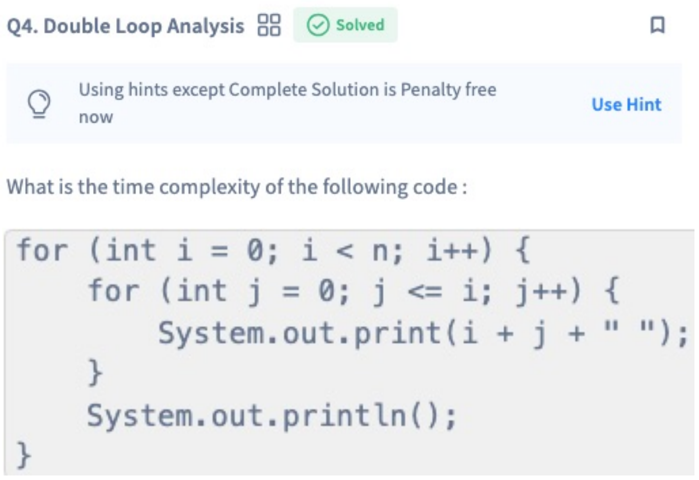
Use Hint

What is the time complexity of the following code :

```
static int func(int n) {
    int s = 0;
    for (int i = 1; i <= 100; i++) {
        s += i;
    }
    return s;
}
```

$\Rightarrow$ 100 iterations

$\Rightarrow O(1)$



Q4. Double Loop Analysis

Solved

Using hints except Complete Solution is Penalty free now

Use Hint

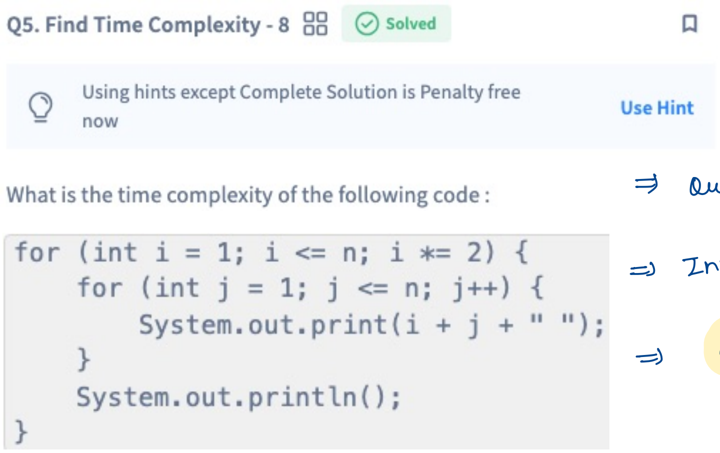
What is the time complexity of the following code :

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

i	j	iterations
0	[0 , 0]	1
1	[0, 1]	2
2	[0, 2]	3
3	[0, 3]	4
⋮	⋮	⋮
n-1	[0, n-1]	n

$\sum$ iterations $= \sum n = \frac{n(n+1)}{2}$

$\Rightarrow O(N^2)$



Q5. Find Time Complexity - 8

Solved

Using hints except Complete Solution is Penalty free now

Use Hint

What is the time complexity of the following code :

```
for (int i = 1; i <= n; i *= 2) {
    for (int j = 1; j <= n; j++) {
        System.out.print(i + j + " ");
    }
    System.out.println();
}
```

$\Rightarrow$ Outer Loop will run $\log_2 N$ times

$\Rightarrow$ Inner Loop will run N times

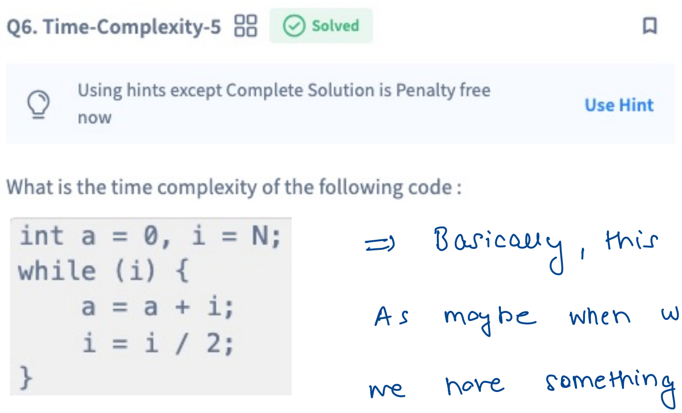
$\Rightarrow O(N\log_2 N)$

Explanation

i	j	iterations
$2^0 = 1$	[1, N]	N
$2^1 = 2$	[1, N)	N
$2^2 = 4$	[1, N]	N
⋮	⋮	⋮
$2^k = N$	[1, N]	N

$\hookrightarrow k = \log_2 N$

$\Rightarrow N + N + N + \dots (\log_2 N \text{ times}) = N\log_2 N$



Q6. Time-Complexity-5 Solved

Using hints except Complete Solution is Penalty free now

Use Hint

What is the time complexity of the following code :

```
int a = 0, i = N;
while (i) {
    a = a + i;
    i = i / 2;
}
```

⇒ Basically, this loop will run till $i > 0$

As maybe when we approach towards the end we have something like

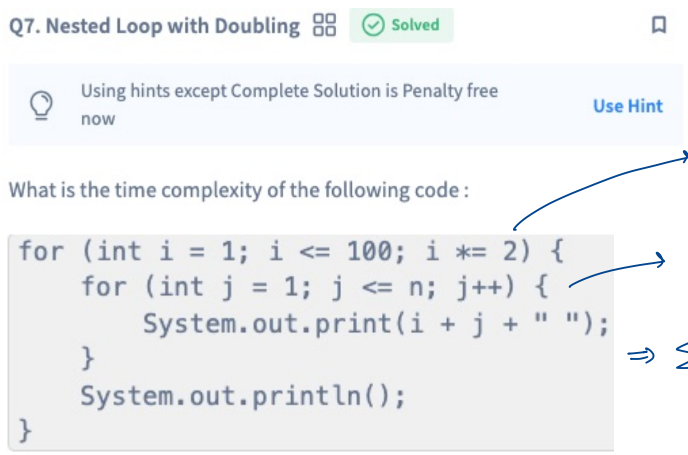
$$\vdots$$

$$i = 3 \xrightarrow{3/2} i = 1 \xrightarrow{1/2} i = 0 \rightarrow \text{loop terminates}$$

⇒ Initially, $i = N$

$$\therefore \; N \times \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} \ldots \; K \text{ times} = 1 \rightarrow \text{after this loop will terminate}$$

$$\frac{N}{2^k} = 1 \Rightarrow \boxed{K = \log_2 N} \quad \therefore \; O = \log_2 N$$



Q7. Nested Loop with Doubling Solved

Using hints except Complete Solution is Penalty free now

Use Hint

What is the time complexity of the following code :

```
for (int i = 1; i <= 100; i *= 2) {
    for (int j = 1; j <= n; j++) {
        System.out.print(i + j + " ");
    }
    System.out.println();
}
```

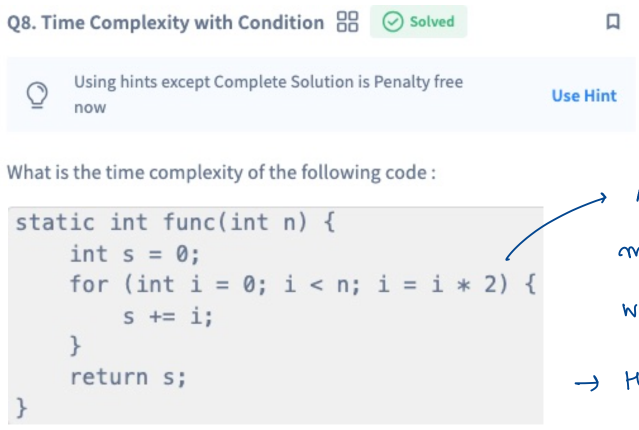
Outer Loop will run $\log_2 100$ times

Inner Loop will run N times

$$\Rightarrow \sum \text{iterations} = \log_2 100 \times N$$

$$= N \underbrace{\log_2 100}_{\text{constant term}}$$

$$\Rightarrow O(N)$$



Q8. Time Complexity with Condition Solved

Using hints except Complete Solution is Penalty free now Use Hint

What is the time complexity of the following code :

```
static int func(int n) {
    int s = 0;
    for (int i = 0; i < n; i = i * 2) {
        s += i;
    }
    return s;
}
```

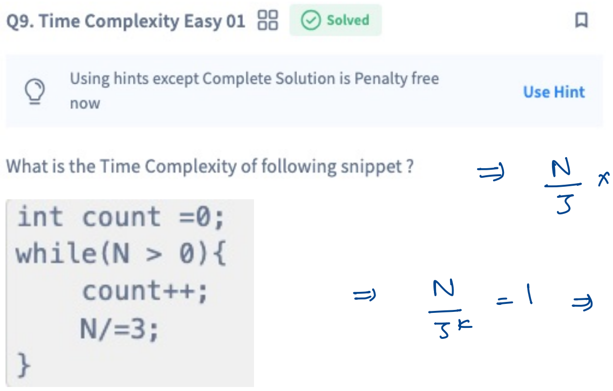
Ah tricky... At first glance it might seem obvious that the loop will run $\log_2 n$ times

→ However, there's a catch!

→ i starts from '0'

→ No amount of multiplication will lead to any change in i

$\Rightarrow$ $O(\infty)$



Q9. Time Complexity Easy 01 Solved

Using hints except Complete Solution is Penalty free now Use Hint

What is the Time Complexity of following snippet ?

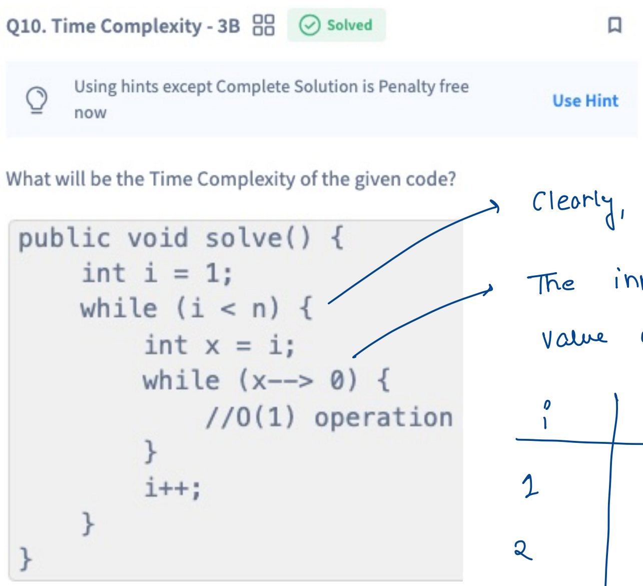
```
int count =0;
while(N > 0){
    count++;
    N/=3;
}
```

$\Rightarrow \frac{N}{3} \times \frac{1}{3} \times \frac{1}{3} \times \ldots$ k times $= 1$

↘ after that $1/3 = 0$ ↓ loop ends

$\Rightarrow \frac{N}{3^k} = 1 \Rightarrow k = \log_3 N$

$\Rightarrow O(\log_3 N)$



Q10. Time Complexity - 3B Solved

Using hints except Complete Solution is Penalty free now

Use Hint

What will be the Time Complexity of the given code?

```
public void solve() {
    int i = 1;
    while (i < n) {
        int x = i;
        while (x--> 0) {
            //O(1) operation
        }
        i++;
    }
}
```

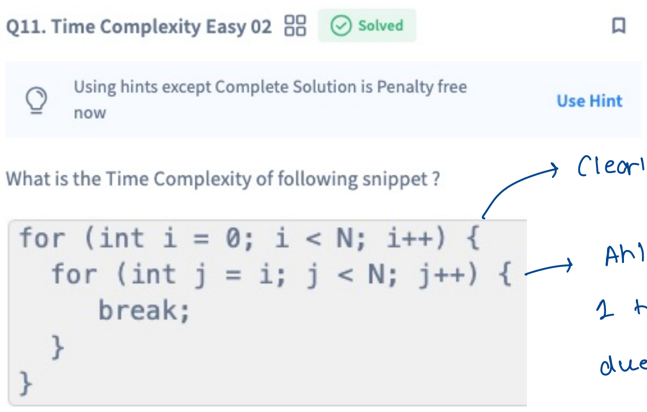
Clearly, outer loop runs N times

The inner loop operates to bring the value of n down to 0

i	n (reverse)	iterations
1	[1,0)	1
2	[2,0)	2
3	[3,0)	3
⋮	⋮	⋮
N	[N,0)	N

$\Rightarrow \sum$ iterations $=$ N + N + … (N times) $=$ N^2 (roughly)

$\Rightarrow$ $O(N^2)$



Q11. Time Complexity Easy 02 Solved

Using hints except Complete Solution is Penalty free now

Use Hint

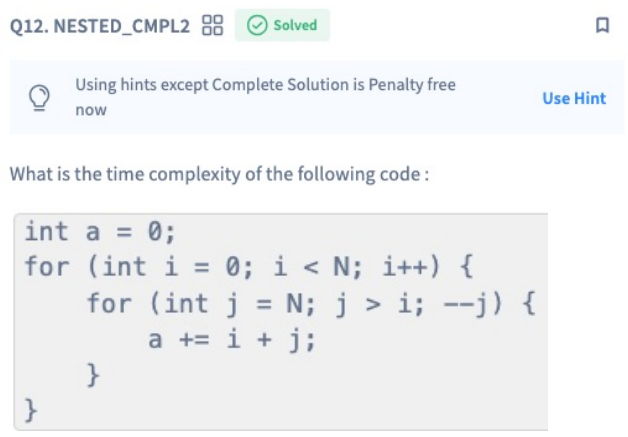
What is the Time Complexity of following snippet ?

```
for (int i = 0; i < N; i++) {
  for (int j = i; j < N; j++) {
     break;
  }
}
```

Clearly outer loop executes N times

Ah! The inner loop only executes 1 time for every outer iteration due to break statement

$\Rightarrow$ $O(N)$


Q12. NESTED_CMPL2 Solved

Using hints except Complete Solution is Penalty free now

Use Hint

What is the time complexity of the following code :

```
int a = 0;
for (int i = 0; i < N; i++) {
    for (int j = N; j > i; --j) {
        a += i + j;
    }
}
```

i	j	iterations
0	[N , 0)	N
1	[N, 1)	N - 1
2	[N, 2)	N - 2
⋮	⋮	⋮
N - 2	[N, N - 2)	2
N - 1	[N, N - 1)	2

$$\Sigma \text{ iterations} = 1 + 2 + \dots + (N-2) + (N-1) + N$$

$$= \frac{N(N+1)}{2} \Rightarrow O(N^2)$$