



Solution Review: Nested Loop with Multiplication (Pro)

This review provides a detailed analysis of the different ways to solve the Nested Loop with Multiplication challenge

We'll cover the following ^

- Solution
 - Time Complexity

Solution

```
1 n = 10 # can be anything
2 sum = 0
3 pie = 3.14
4 j = 1
5 for var in range(n):
6     while j < var:
7         sum += 1
8         j *= 2
9     print(sum)
10
```



The outer loop in the main function has n iterations as it iterates on the list generated from `range(n)`. If the condition `j < var` is true, the inner loop entered. However, immediately, `j` is doubled. Note that `j` is not reset to 1 in

the code. The inner while loop will run at most once for each iteration of the outer loop. Therefore, **lines 6, 7 and 8** run $O(n)$ times each. Since we are interested in an upper bound on the worst case running time, let's assume these statements run exactly n times.



Statement	Number of Executions
<code>n = 10</code>	1
<code>sum = 0</code>	1
<code>pie = 3.14</code>	1
<code>for var in range(n):</code>	n
<code>while j < var:</code>	n
<code>sum+=1;</code>	n
<code>j *= 2;</code>	n
<code>print(sum)</code>	n

Time Complexity#

Running Time Complexity = $1 + 1 + 1 + n + n + n + n + n$

$$= 3 + 5n$$



To find the Big O time complexity,



1. Drop the leading constants $\Rightarrow n$

2. Drop the lower order terms $\Rightarrow n$

The Big O time complexity of the above is hence, $O(n)$

Interviewing soon? We've partnered with Hired so that companies apply to you instead of you applying to them. [See how](#) ⓘ



← Back

Next →

Challenge 7: Nested Loop with Multipl...

Complexity Quiz: Test your understand...

✓ Completed



Report an Issue

