

WEEK 4 - Task A.1

Show how to calculate the Big O notation for the segment of C# code shown below

Solution: Comments indicate the thought and logic. Let, C = number of times the line is executed.

```
int n = 50; // C = 1

int r = 20; // C = 1

int dummy = 0; // C = 1

Console.WriteLine("Creating a nested loop"); // C = 1

for (int i = 0; i < n; i++) { // C = n + 1

    Console.WriteLine("Inside the first loop"); // C = n

    Dummy++; // C = n

    for (int j = 0; j < n; j++) { // C = n * (n+1)

        r = r + dummy; // C = n * n = n^2

        Console.WriteLine("Inside the second loop " + r); // C = n * n = n^2

    }

}
```

Each for Loop is executed **n+1** times while the body is executed **n** times. Since we have a nested for loop - each new nest is executed n to the power of the times of the for loop. In this case the body of the “Second” for loop is run **n^2 times**.

Now - we Add all the C values.	=> $\text{Big}(O) = 1 + 1 + 1 + 1 + (n+1) + n + n + (n * (n+1)) + n^2 + n^2$
(After Simplifying we get)	=> $\text{Big}(O) = 5 + 4n + 3n^2$
(Eliminate the constant)	=> $\text{Big}(O) = 4n + 3n^2$
(Eliminate the coefficients)	=> $\text{Big}(O) = n + n^2$
(Consider the largest significant figure)	=> $\text{Big}(O) = n^2$

Therefore - The BigO notation for the sample code provided is **n^2**.