20.15. TIME COMPLEXITY CALCULATION NESTED FOR LOOP (MORE THAN TWO LOOP).

EXAMPLE 1

```
for(i = 1; i ≤ n; i + +){
        for(j = 1; j ≤ n; j + +){
            for(k = 1; k ≤ n; k + +){
                  c = c + 1;
            }
      }
}
```

ANSWER

The loop runs like:

$f(1) \leq n, when\ i = 1$

$f(1) \leq n, when\ j = 1$

$f(1) \leq n, when\ k = 1$

$\mathrm{c} = \mathrm{c} + 1 - - - ---\rightarrow (1)$

$f(2) \leq n, when\ k = 2$

$\mathrm{c} = \mathrm{c} + 1 - - - ---\rightarrow (2)$

….

$f(n) \leq n, when\ k = n$

$\mathrm{c} = \mathrm{c} + 1 - - - ---\rightarrow (n)$

$$T(n) = \sum_{i=1}^{n} 1 = (1 + 1 + \cdots n\ times) = n$$

$f(2) \leq n, when\ j = 2$

$f(1) \leq n, when\ k = 1$

$\mathrm{c} = \mathrm{c} + 1 - - - ---\rightarrow (1)$

$f(2) \leq n, when\ k = 2$

$\mathrm{c} = \mathrm{c} + 1 - - - ---\rightarrow (2)$

….

$f(n) \leq n, when\ k = n$

$\mathrm{c} = \mathrm{c} + 1 - - - ---\rightarrow (n)$

$$T(n) = \sum_{i=1}^{n} 1 = (1 + 1 + \cdots n\ times) = n$$

……

$f(n) \leq n, when\ j = 2$

$f(1) \leq n, when\ k = 1$

$\mathrm{c} = \mathrm{c} + 1 - - - ---\rightarrow (1)$

$f(2) \leq n, when\ k = 2$

$\mathrm{c} = \mathrm{c} + \mathbf{1} - - - - -- \rightarrow (\mathbf{2})$

....

$\boldsymbol{f(n) \leq n\,, when\; k = n}$

$\mathrm{c} = \mathrm{c} + \mathbf{1} - - - - -- \rightarrow (\boldsymbol{n})$

$$\boldsymbol{T(n) = \sum_{i=1}^{n} 1 = (1 + 1 + \cdots n\; times) = n}$$

$\boldsymbol{And\; j\; runs\; upto\; n\; times\; produces}$:

$$\boldsymbol{T(n) = \sum_{i=1}^{n} n = (n + n + \cdots n\; times) = n^2}$$

$\boldsymbol{f(n) \leq n\,, when\; i = n}$

$\boldsymbol{f(1) \leq n\,, when\; j = 1}$

$\boldsymbol{f(1) \leq n\,, when\; k = 1}$

$\mathrm{c} = \mathrm{c} + \mathbf{1} - - - - -- \rightarrow (\mathbf{1})$

$\boldsymbol{f(2) \leq n\,, when\; k = 2}$

$\mathrm{c} = \mathrm{c} + \mathbf{1} - - - - -- \rightarrow (\mathbf{2})$

....

$\boldsymbol{f(n) \leq n\,, when\; k = n}$

$\mathrm{c} = \mathrm{c} + \mathbf{1} - - - - -- \rightarrow (\boldsymbol{n})$

$$\boldsymbol{T(n) = \sum_{i=1}^{n} 1 = (1 + 1 + \cdots n\; times) = n}$$

$\boldsymbol{f(2) \leq n\,, when\; j = 2}$

$\boldsymbol{f(1) \leq n\,, when\; k = 1}$

$\mathrm{c}= \mathrm{c} + \mathbf{1} - - - - - - \rightarrow (\mathbf{1})$

$\boldsymbol{f(2) \leq n\,, when\; k = 2}$

$\mathrm{c} = \mathrm{c} + \mathbf{1} - - - - - - \rightarrow (\mathbf{2})$

....

$\boldsymbol{f(n) \leq n\,, when\; k = n}$

$\mathrm{c} = \mathrm{c} + \mathbf{1} - - - - - - \rightarrow (\boldsymbol{n})$

$$\boldsymbol{T(n) = \sum_{i=1}^{n} 1 = (1 + 1 + \cdots n\; times) = n}$$

......

$\boldsymbol{f(n) \leq n\,, when\; j = 2}$

$\boldsymbol{f(1) \leq n\,, when\; k = 1}$

$\mathrm{c}= \mathrm{c} + \mathbf{1} - - - - - - \rightarrow (\mathbf{1})$

$\boldsymbol{f(2) \leq n\,, when\; k = 2}$

$\mathrm{c} = \mathrm{c} + \mathbf{1} - - - - - - \rightarrow (\mathbf{2})$

....

$\boldsymbol{f(n) \leq n\,, when\; k = n}$

$\mathrm{c} = \mathrm{c} + \mathbf{1} - - - - - - \rightarrow (\boldsymbol{n})$

$$\boldsymbol{T(n) = \sum_{i=1}^{n} 1 = (1 + 1 + \cdots n\; times) = n}$$

$\boldsymbol{And\; j\; runs\; upto\; n\; times\; produces}$:

$$\boldsymbol{T(n) = \sum n = (n + n + n + \cdots + n) = n^2}$$

$\boldsymbol{And\ i\ runs\ upto\ n\ times\ produces}$:

$$T(n^2) = \sum n^2 = (\,n^2 + n^2 + \cdots + n\ times) = n \times n^2 = n^3$$

$\boldsymbol{Hence\ the\ above\ loop\ runs\ up\ to\ n^3\ times\ and}$
$\mathrm{c} = \mathrm{c} + \mathbf{1}$, $\boldsymbol{prints\ n^3\ times} \Longrightarrow \boldsymbol{O\left(n^3\right)}$.