Rafiul Hassan.

Big O

⇒ Time Function ⇒ $T(n) = 5n^2 + n + 4$.

Find constant ⇒ $c = ?$

⇒ $N = ?$

Big O ⇒ $f(n) \leq c * g(n)$, for $n \geq N$

Solution:-

(i) $T(n) = 5n^2 + n + 4$ — bign

$5n^2 + n + 4 \leq 5n^2 + n^2 + 4n^2$ — multiply n^2 →

$5n^2 + n + 4 \leq 10n^2$ →

∴ By the equation of Big O Rule ⇒ $c = 10$

$n \geq 1$

∴ $N = 1$.

⇒ Find Big O

$T(n) = 2n^3 + n^2 + 6n + 12$

Find constant $c = ?$

⇒ $N = ?$

solution:-

(i) $2n^3 + n^2 + 6n + 12 \leq 2n^3 + n^3 + 6n^3 + 12n^3$ — bign, multiply n^3

⇒ $2n^3 + n^2 + 6n + 12 \leq 21n^3$

∴ As per Big O ⇒ $c = 21$

$n \geq 1$

$N = 1$ //

Rafiul Hassan.

Big O following Code.

⇒ First for loop runs of (n) time
so O(n)

⇒ Second nested for loop runs for i and j
so we know nested loop ⇒ $n \times n = n^2$

⇒ Hello runs one operation ⇒ O(1)

∴ (first loop) ⟶ O(n)
(second loop) i x j
$n \times n = n^2$ $O(n^2)$
Hello ⟶ $O(1)$
∴ Big O is $O(n^2)$
for nested loops running Hello.