function $x = f(n)$

$x = 1$ $\Rightarrow \Theta(1) \Rightarrow c$

for $i = 1:n$ $\rightarrow \sum_{i=1}^{n} 1$

for $j = 1:n$ $\rightarrow \sum_{j=1}^{n} 1$

$x = x+1$ $\Rightarrow \Theta(1) \Rightarrow c$

1. Runtime: $T(n) = \sum_{i=1}^{n} \sum_{j=1}^{n} 1$

$= \sum_{i=1}^{n} n-1+1$

$= n \sum_{i=1}^{n} 1 \Rightarrow n(n-1+1)$

$= n^2$

$\therefore$ Runtime of the algorithm i.e $\Theta = n^2$

2. Check the attached graph

3. Big - Theta (i.e) the Runtime $= n^2 \Rightarrow \Theta(n^2)$

$f(n)$

Big - O : $0 \le f(n) \le c\, g(n)$

$\therefore\ n^2 \le c \cdot n^2$ — ①

Take $c = 2$ & substitute in ①

$n^2 \le 2 \cdot n^2 \Rightarrow 2n^2$ $O(n^2)$

Big - Ω : $0 \le c\, g(n) \le f(n)$

$\therefore\ c \cdot n^2 \le n^2$ — ②

Take $c = 1/2$ & substitute in ②

$\frac{1}{2} \cdot n^2 \le n^2 \Rightarrow n^2/2$ $\Omega(n^2)$

Big $\Theta \Rightarrow n^2 \Rightarrow \Theta(n^2)$

Observation: The best case, average case & worst case of this algorithm is all same since it executes n times any way.

4. $n_0 = 8000$ $\Rightarrow$ Reason: rapid change in curve

$n > n_0$: Below $n_0 \Rightarrow$ polynomial trend is not there (Check in graph)

Algorithm changed

$x = 1$ $\rightarrow c_1$

$y = 1$ $\rightarrow c_2$

for $i = 1:n$ $\rightarrow \sum_{i=1}^{n} 1$

for $j = 1:n$ $\rightarrow \sum_{j=1}^{n} 1$

$x = x+1$ $\rightarrow c_3$

$y = i + j$ $\rightarrow c_4$

4. The time taken has increased by an average of 10 seconds

5. No, the runtime of this algorithm is still $\Theta(n^2)$

$$c_1 + c_2 + \sum_{i=1}^{n} 1 + \sum_{j=1}^{n} 1 + c_3 + c_4$$

$$= c + \sum_{i=1}^{n} \sum_{j=1}^{n} 1$$

$$= c + \sum_{i=1}^{n} (n-1+1) \Rightarrow c + n \sum_{i=1}^{n} 1$$

$$= c + n(n-1+1)$$

$$= c + n^2 \quad \therefore \Theta = n^2 \quad \Theta(n^2)$$