Hands - On 3

function : $x = f(n)$

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

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

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

$x = x+1;$

1) Find the runtime of the algorithm mathematically:

Sol: outer loop runs from 1 to n

And for each iteration of outer loop, the inner loop also runs from 1 to n. So, the total no. of iterations for both loops together is sum of no's from 1 to n for each loop.

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

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

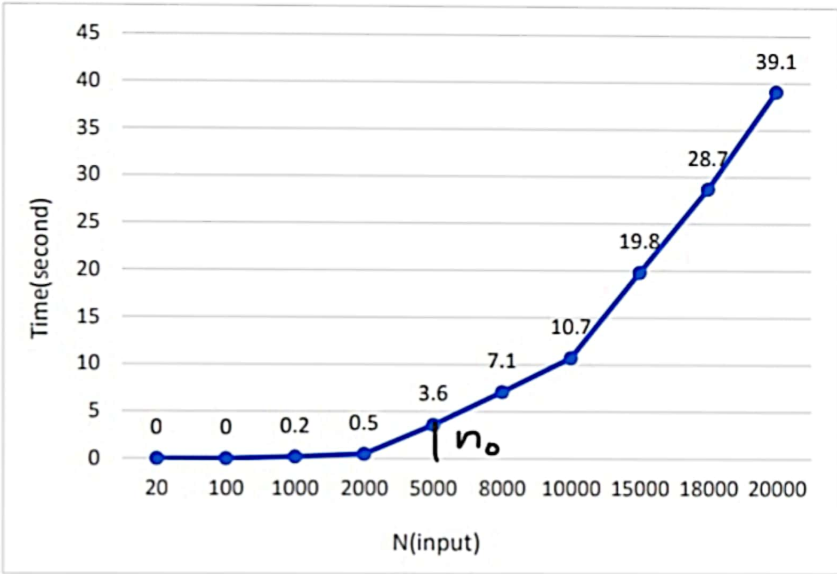
$$\Rightarrow n \sum_{i=1}^{n} 1 \Rightarrow n(n - \cancel{1} + \cancel{1})$$

$$\Rightarrow T(n) = n^2$$

$$\Rightarrow T(n) = \Theta(n^2)$$

2) Please find the attached graph.

X-Axis	Y-Axis
N(input)	Time(seconds)
20	0
100	0
1000	0.2
2000	0.5
5000	3.6
8000	7.1
10000	10.7
15000	19.8
18000	28.7
20000	39.1



for modified function

X-Axis	Y-Axis
N(input)	Time(seconds)
20	0
100	0
1000	0.6
2000	1.3
5000	4.2
8000	8.9
10000	14.8
15000	29.3
18000	38.6
20000	58.4

3) Big theta $\Rightarrow$ the runtime $= n^2$

from f(n) $\Rightarrow \Theta(n^2)$

Big-O $\Rightarrow$ $0 \le f(n) \le cg(n)$

$\because f(n) = n^2$

$\Rightarrow n^2 \le c \cdot n^2$

Let us consider $c = 4$ & substitute in above

$n^2 \le 4n^2 \Rightarrow 4n^2 \Rightarrow O(n^2)$

Big Ω $\Rightarrow$ $0 \le cg(n) \le f(n)$

$\because f(n) = n^2$

$\Rightarrow c \cdot n^2 \le n^2$

Let us consider $c = 1/2$ & substitute above

$1/2\, n^2 \le n^2 \Rightarrow \frac{n^2}{2} \le n^2$

$\Rightarrow \frac{n^2}{2} \Rightarrow \Omega(n^2)$

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

$\therefore$ From the above calculations, we can see that

Big-O, Big-Ω, and Big Θ are same. ie;

The best case, average case & worst case of this function is all same, since it executes n times.

4) $n_0 = 5000$

There is a rapid change in curve when $n = 5000$

So, $n_0 = 5000$

$\because n \ge n_0$: Below n_0 the polynomial there is not specified and polynomial trend not followed.

Modification of function:

```
x = f(n)
x = 1 ;
y = 1 ;
for i = 1 : n .
  for j = 1 : n
    x = x + 1 ;
    y = i + j
```

4) we can see that the time taken has been increased slightly with an average of 13 seconds

5) The runtime of the modified function is

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

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

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

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

$$\Rightarrow c + n(n - \cancel{1} + \cancel{1})$$

$$\Rightarrow c + n^2$$

$$\therefore \Theta\, n^2 \Rightarrow T(n) = \Theta(n^2)$$

$\therefore$ It does not effect the results from #1. The runtime of the modified function is still $\Theta(n^2)$