Design & Analysis of Algorithms. Name : Patibandla Hemanth Kumar

Hands-on 3

UTA ID :- 1002207795

---

① Function $x = f(n)$

$x = 1;$

for $i = 1:n$

for $j = 1:n$

$x = x + 1;$

1) Find runtime of Algorithm mathematically

Functions

$x = 1;$ ⇒ runtime is # 1

for $i = 1:n$ ⇒ runtime is # $n+1$

for $j = 1:n$ ⇒ runtime is # $n(n+1)$

$x = x+1;$ ⇒ run time is # n^2

Therefore it can be calculated by using summation formula.

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

After solving we get

$$T(n) = 1 + n + 1 + n^2 + n + n^2$$

$$T(n) = 2n^2 + 2n + 2$$

Finally runtime for Algorithm is

$T(n) = 2n^2 + 2n + 2$.

(3) Find polynomials that are upper and lower bounds on your Curve from #2. From this specify a big O, a bigomega and what big-theta is.

Ans) Here we can take $C1 = \frac{1}{2}$ & $C2 = 5$ Since

$\frac{1}{2}n^2 \leq 2n^2 + 3n + 2 \leq 5n^2 (1)$

the following equation will get us the following notation

$f(n) = O(n^2)$ ⇒ this is big-o

$f(n) = \Omega(n^2)$ ⇒ this is big-omega.

$f(n) = \Theta(n^2)$ ⇒ this is big-Theta.

(4) Find the approximate (eye ball it) location of "n-0". Do this by zooming in on your plot and indicating on the plot where n-0 is & why you picked this value.

Ans) Here we can say that the "n-0" is 1 and also we can say that $n \geq 2$ is inequality.

$n\_0 = 1$ because all values are ≥ 1

∴ $T(n) = 2n^2 + 2n + 2$ with upper and lower bounds

Here we can state that at $x = 1, 4$ upper bound is larger than $T(n)$

Because of which n_0 will be 2, is 1st integer.

(5) Will this increase how long it takes the algorithm to run.

Ans) Here It is given that

$x=1;$ #1 $y=1;$ #1 for $i=1:n$ #n+1 for $j=1:n$ #n(n+1) $x=x+1;$

#n^2 $y=i+j;$ #n^2

is changed

$x = f(n)$

$x=1 \Rightarrow C_1$ (Cost) $\Rightarrow 1$ (Time)

$y=1 \Rightarrow C_2$ (Cost) $\Rightarrow 1$ (Time)

for $i=1:n. \Rightarrow C_3$ (Cost) $\Rightarrow \sum_{i=1}^{n} 1$

for $j=1:n \Rightarrow C_4$ (Cost) $\Rightarrow \sum_{i=1}^{n} \sum_{j=1}^{n}$

$x=x+1 \Rightarrow C_5$ (Cost) $\Rightarrow \sum_{i=1}^{n} \sum_{j=1}^{n} 1$

$y=i+j \Rightarrow C_6$ (Cost) $\Rightarrow \sum_{i=1}^{n} \sum_{j=1}^{n} 1$

From this $T(n) = C_1 + C_2 + C_3 \sum_{i=1}^{n} 1 + (C_4 + C_5 + C_6)\left(\sum_{i=1}^{n} \sum_{j=1}^{n} 1\right)$

$= (C_1 + C_2) + C_3 n + (C_4 + C_5 + C_6) n^2$

(6) Will it effect your result from #1?

Ans) Here actually the whole time complexity will be same. is $O(n^2)$. Here step functions increased. Constant values is changed & the structure of $T(n)$ is not changed so,

No it will not effect result from #1