1/28/19

HomeWork # 1

1. Let $f(n)$ and $g(n)$ be positive functions. Prove by the definition of Θ that $F(n) = \Theta(G(n))$ where $F(n) = \max(f(n), g(n))$ and $G(n) = f(n) + g(n)$

A.

Since $f(n)$ and $g(n)$ are positive functions, we can say that,

$$f(n) \leq f(n) + g(n) \quad \text{(1)} \quad \text{and} \quad g(n) \leq f(n) + g(n) \quad \text{(2)}$$

Summing 1, 2, we get,

$$f(n) + g(n) \leq 2(f(n) + g(n))$$

Consider, $F(n) = \Theta(G(n))$ is true,

$$\max(f(n), g(n)) = \Theta(f(n) + g(n))$$

$$\Rightarrow \quad f(n) + g(n) \leq 2\max(f(n)$$

We know that for Ω-notation,

$f(n) = \Omega(g(n))$ when $f(n) \geq c \cdot g(n) \ \forall n \geq n_0$

$\therefore \max(f(n), g(n)) = \Omega(f(n)+g(n))$

So, $F(n) = \Theta(f(n)+g(n))$

given $G(n) = f(n) + g(n)$

$\therefore F(n) = \Theta(G(n))$

2. An array S[·] stores n distinct numbers such that the first k numbers are smaller than x and the other numbers are larger than x. Suppose that INSERTION SORT is used to sort S. The number of shifts t made by INSERTION SORT depends on S. Find the largest value of t as a function of n and k. Argue why.

A. Given that array S is divided into two parts – 'k' and 'n-k'. The first k elements contains values smaller than x. and other part contains values larger than x.

Thus we can conclude that first part shift is independent of second part sorting. So, divide the problem into two parts.

* For insertion sort, worst case is,

$$\frac{n(n-1)}{2}$$

So, for the first k numbers,

$$t_k = \frac{k(k-1)}{2}$$

and for next n-k numbers,

$$t_{n-k} = \frac{(n-k)(n-k-1)}{2}$$

So, total shifts for array s[ ] is,

$$t = t_k + t_{n-k}$$

$$= \frac{k(k-1) + (n-k)(n-k-1)}{2}$$

3. ALG1(n)

```
S = 0;
for(i=1; i<=n; i++)
    for (j=1;j<=n;j+=i)
        S+=j*j
```

A. In the given for loops,

for(i=1; i<=n; i++) – Runs for 'n' times

for (j=1;j<=n;j+=i)

– This loop runs for $\lceil n/i \rceil$ times each iteration.

for eg: Take n = 6

i	j values
1	1 2 3 4 5 6
2	1 3 5
3	1 4
4	1 5
5	1 6
6	1

So ~~fo~~ the total cost is,

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

$$= n^2 \sum_{i=1}^{n} \frac{1}{i}$$

$$= n \log^n \left(\frac{2 + 2n - 1}{2 - 1} \right)$$

$$= n\ \cancel{\log} \left(2n + 1 \right)$$

In Harmonic progression, sum of 'n' numbers is,

$$S_n = \frac{1}{d} \ln \left[\frac{2a + (2n-1)}{2a - d} \right]$$

$\therefore$ The asymptotic analysis of the running time is $\Theta(n \ln n)$.

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

5. Let $f(n)$ and $g(n)$ be asymptotically positive functions. Prove or disaprove.

a) $f(n) = O(g(n))$ implies $g(n) = O(f(n))$

FALSE

Let $f(n) = n^2$ and $g(n) = \log n$

Then $g(n) = O(f(n))$, but $f(n) \neq O(g(n))$

b) $f(n) = O(g(n))$ implies $2^{f(n)} = O(2^{g(n)})$

FALSE

Let $f(n) = n$ and $g(n) = n/2$,

then $f(n) = O(g(n))$, when $c \geq 2$

Given, $\lim_{n \to \infty} \frac{2^{n/2}}{2^n} < 1$

$\therefore 2^{f(n)} \neq O(2^{g(n)})$

4. Given,

$(\sqrt{2})^{\lg n}$ n^2 $n!$ $\ln n$ $(\frac{3}{2})n$ n^3

$\lg^2 n$ $\lg(n!)$ 2^{2^n} $n.\lg n$ $\lg\lg n$ $n.2^n$

$4^{\lg n}$ $(n+1)!$ n 2^n $2^{\lg n}$ e^n

A. $2^{2^n} > (n+1)! > n! > e^n > n.2^n > 2^n$

$(\frac{3}{2})n > n^3 > n^2, 4^{\log n}, n.\lg n, \lg(n!)$

$n, 2^{\lg n} > \sqrt{2}^{\lg n} > \lg^2 n > \ln n > \lg^2 n$

$> \ln n > \lg \lg n$

Scanned by CamScanner