

HW #2

$$2w\left(\frac{n}{3}\right) + 1$$

$$c(\text{root}) = 1$$

$$c(\text{level } 1) = 1\left(\frac{n}{3}\right) + 1\left(\frac{n}{3}\right)$$

$$1) a) O(n^{\log_3 2})$$

$$b) \text{ ~~5~~ } w(n) = 5w\left(\frac{n}{4}\right) + n$$

$$c(\text{root}) = n$$

$$c(\text{level}) = \left(\frac{n}{4}\right) + \left(\frac{n}{4}\right) + \left(\frac{n}{4}\right) + \left(\frac{n}{4}\right) + \left(\frac{n}{4}\right)$$

$$O(n^{\log_4 5})$$

$$c) \text{ ~~7~~ } w(n) = 7w\left(\frac{n}{7}\right) + n$$

$$c(\text{root}) = n$$

$$c(\text{level } 1) = 7\left(\frac{n}{7}\right) = n$$

$$O(n \log_b a \cdot \log_b 7) = O(n \log n)$$

$$d) \text{ ~~9~~ } w(n) = 9w\left(\frac{n}{3}\right) + n^2$$

$$c(\text{root}) = n^2$$

$$c(\text{level } 1) = 9\left(\frac{n}{3}\right)^2 \Rightarrow 9\left(\frac{n^2}{9}\right) = n^2$$

$$\begin{aligned} O(n \log_b a \cdot \log_b n) &= O(n \log_3 9 \cdot \log_3 n) \\ &= O(n^2 \log n) \\ &= O(n^2 \log n) \end{aligned}$$

$$e) w(n) = 8w\left(\frac{n}{2}\right) + n^3$$

$$c(\text{root}) = n^3$$

$$c(\text{level } 1) = 8\left(\frac{n}{2}\right)^3 = 8\left(\frac{n^3}{8}\right) = n^3$$

$$O(n \log_2 8 \cdot \log_2 2^n)$$

$$O(n^3 \log_2 n)$$

$$O(n^3 \log n)$$

$$f) w(n) = 49w\left(\frac{n}{25}\right) + n^{3/2} \log n$$

$$c(\text{root}) = n^{3/2} \log n$$

$$c(\text{level } 1) = 49\left(\frac{n}{25}\right)^{3/2} \log\left(\frac{n}{25}\right)$$

$$f(n) = n^{3/2} \log n$$

$$-b: 25$$

$$a: 49$$

$$f(n) = \Theta(n^c \log^k n) \quad c = \log_{25} 49$$

g) So, by master theorem

$$w(n) \in O(n^{\log_{25} 49} \log^2 n)$$

$$g) w(n-1) + 2 = w(n)$$

$$w(n-1) = w(n-2) + 2$$

$$w(n-2) = w(n-3) + 2$$

$$w(n) = 2k + w(n-k)$$

$$k = n-1$$

$$w(n) = 2(n-1) + 1$$

$$\in O(n)$$

$$n) \quad w(n-1) + n^c = w(n) \quad c \geq 1$$

$$w(n-1) = w(n-2) + (n-1)^c$$

$$w(n) = w(n-2) + (n-1)^c + n^c$$

$$w(n) = w(n-3) + (n-2)^c + (n-1)^c + n^c$$

$$= w(n-k) + \dots + (n-k+1)^c$$

$$\text{Let } k = n-1$$

$$= w(1) + n^c + (n-1)^c + \dots + (n-k+1)^c$$

$$= 1 + n^c + (n-1)^c + \dots + 1^c$$

$$\leq 1 + n^c + n^c + \dots + n^c$$

$$= 1 + n^{c+1}$$

$$\in O(n^{c+1}) \text{ for any } c \geq 1$$

2.)

$$i) \quad w(n) = w(\sqrt{n}) + 1$$

$$w(2) = 1$$

$$w(4) = w(\sqrt{4}) + 1 = w(2) + 1 = 2$$

$$w(16) = w(\sqrt{16}) + 1 = w(4) + 1$$

$$w(2^2) = w(2) + 1$$

$$w(2^4) = w(2^2) + 1$$

$$w(2^8) = w(2^4) + 1$$

$$\rightarrow w(2^{2^k}) = w(2^{2^{k-1}}) + 1$$

$$\text{Let } n = 2^{2^k}$$

$$w(n) = w(\sqrt{n}) + 1$$

$$n = 2^{2^k}$$

$$\log n = 2^k$$

$$\log(\log n) = k$$

$$O(\log(\log n))$$

2) Alg A: $5W(\frac{n}{2}) + n = w(n)$

Alg B: $2W(n-1) + 1 = w(n)$

Alg C: $9W(\frac{n}{3}) + n^2 = w(n)$

A: $w(n) \in O(n^{\log_2 5})$

B: $w(n) \in O(n)$

C: $w(n) \in O(n^2 \log n)$

I would choose algorithm b because
 $n < n^{\log_2 5} \leq n^2 \log n$ for a sufficiently
large n