

CS 251 Homework 3

Dwijen Chawra (dchawra)

October 19, 2023

Problem 0.

Problem 1:

Resources used: None

Collaborators: None

Problem 2:

Resources used: None

Collaborators: None

Problem 3:

Resources used: None

Collaborators: None

Problem 4:

Resources used: None

Collaborators: None

Problem 5:

Resources used: None

Collaborators: None

Problem 1.

(a) Simplifying the original functions to their Big O notation:

- (i) $\log(2^{4n}) = 4n = O(n)$
- (ii) $\log^2 4n^2 = (2 + 2\log n)^2 = O(\log^2 n)$
- (iii) $2^{10 \log n} = n^{10} = O(n^{10})$
- (iv) $\log_{10}(n^{12}) = 12 \log_{10} n = O(\log_{10} n)$
- (v) $4^{\log_2 n^2} = n^4 = O(n^4)$
- (vi) $4^n = O(4^n)$
- (vii) $\sqrt{8n} = O(\sqrt{n})$
- (viii) $\binom{n}{2} = \frac{n(n-1)}{2} = O(n^2)$
- (ix) $n^2 / \log n = O(n^2 / \log n)$
- (x) $4^{2n} = O(16^n)$
- (xi) $n^2 + 64n^{1.89} = O(n^2)$

The functions ordered are as follows:

- (i) $\log_{10}(n^{12})$
- (ii) $\log^2 4n^2$
- (iii) $\sqrt{8n}$
- (iv) $\log(2^{4n})$
- (v) $n^2 / \log n$
- (vi) $\binom{n}{2}$ and $n^2 + 64n^{1.89}$
- (vii) $4^{\log n^2}$
- (viii) $2^{10 \log n}$
- (ix) 4^n
- (x) 4^{2n}

(b) **For i:** $f(n) = (\log 4n^2)^2$ vs. $g(n) = \log_{10} n^{12}$

$f(n)$ simplifies to $(\log 4 + \log n^2)^2 = (2 + 2\log n)^2 = 4\log n^2 + 8\log n + 4$

$g(n)$ simplifies to $\frac{12\log n}{\log 10}$

By raising the lower terms to the highest power for $f(n)$ we get

$$f(n) = 16\log^2 n \text{ vs. } g(n) = \frac{12\log n}{\log 10}$$

Using $C = 1$, and $k = 1$, we get $16\log^2 n \leq \frac{12\log n}{\log 10}$ for all $n \geq 1$.

This example is invalid, and as n approaches infinity, $g(n)$ is never greater than $f(n)$. The fraction term in $g(n)$ approaches 0 as n approaches infinity. Therefore $g(n)$ cannot be a bound for $f(n)$.

For ii)

$$\binom{f(n)=n}{2} \text{ vs. } g(n) = n^2/\log_2 n$$

Using $C = 1$, and $k = 8$, we get $\frac{n(n-1)}{2} \leq \frac{n^2}{\log_2 n}$ for all $n \geq 8$.

If we compare the behavior at infinity, the log term in g causes the function to approach 0, which means that $g(n)$ is never greater than $f(n)$ for all $n \geq 8$. This means that $g(n)$ is not a Big O bound for $f(n)$.

For iii)

$$f(n) = 4^n \text{ vs. } g(n) = 4^{2n}$$

Using $C = 1$, and $k = 1$, we get $4^n \leq 4^{2n}$ for all $n \geq 1$.

Therefore, $f(n) = O(g(n))$.

Problem 2.

(a)

$$\sum_{i=1}^{n^2} \sum_{j=1}^i \sum_{k=1}^{\lceil n/8 \rceil} 1 \quad (1)$$

$$= \sum_{i=1}^{n^2} \sum_{j=1}^i \lceil n/8 \rceil \quad (2)$$

$$= \sum_{i=1}^{n^2} \lceil n/8 \rceil \sum_{j=1}^i 1 \quad (3)$$

$$= \sum_{i=1}^{n^2} \lceil n/8 \rceil i \quad (4)$$

$$\text{Let } n^2 = k \quad (5)$$

$$= \sum_{i=1}^k \lceil n/8 \rceil i \quad (6)$$

$$= \lceil n/8 \rceil \sum_{i=1}^k i \quad (7)$$

$$= \lceil n/8 \rceil \frac{k(k+1)}{2} \quad (8)$$

$$= \lceil n/8 \rceil \frac{n^2(n^2+1)}{2} \quad (9)$$

$$= \frac{n^2(n^2+1)}{2} \lceil n/8 \rceil \quad (10)$$

$$= \frac{n^4 + n^2}{2} \lceil n/8 \rceil \quad (11)$$

(b)

$$\sum_{i=1}^n \sum_{j=1}^i \sum_{k=j}^n 3 \quad (12)$$

$$= \sum_{i=1}^n \sum_{j=1}^i 3(n-j+1) \quad (13)$$

$$= \sum_{i=1}^n 3 \sum_{j=1}^i (n-j+1) \quad (14)$$

$$= \sum_{i=1}^n 3 \left(\sum_{j=1}^i n - \sum_{j=1}^i j + \sum_{j=1}^i 1 \right) \quad (15)$$

$$= \sum_{i=1}^n 3 \left(ni - \frac{i(i+1)}{2} + i \right) \quad (16)$$

$$= 3 \sum_{i=1}^n \left(ni - \frac{i(i+1)}{2} + i \right) \quad (17)$$

$$= 3 \left(\sum_{i=1}^n ni + \sum_{i=1}^n i - \sum_{i=1}^n \frac{i(i+1)}{2} \right) \quad (18)$$

$$= 3 \left(n \sum_{i=1}^n i + \sum_{i=1}^n i - \frac{1}{2} \sum_{i=1}^n i^2 - \frac{1}{2} \sum_{i=1}^n i \right) \quad (19)$$

$$= 3 \left(\frac{n^2(n+1)}{2} + \frac{n(n+1)}{2} - \frac{n(n+1)(2n+1)}{12} - \frac{3n(n+1)}{12} \right) \quad (20)$$

$$= 3 \left(\frac{n^3 + n^2}{2} - \frac{1}{2} \left(\frac{2n^3 + 3n^2 + n}{6} + \frac{n^2 + n}{2} \right) + \frac{n^2 + n}{2} \right) \quad (21)$$

$$= 3 \left(\frac{2n^3 + 2n^2}{4} - \frac{2n^3 + 3n^2 + n}{12} - \frac{n^2 + n}{4} + \frac{2n^2 + 2n}{4} \right) \quad (22)$$

$$= 3 \left(\frac{2n^3 + 4n^2 + 2n}{4} - \frac{2n^3 + 3n^2 + n}{12} - \frac{n^2 + n}{4} \right) \quad (23)$$

$$= \frac{6n^3 + 9n^2 + 3n}{4} - \frac{2n^3 + 3n^2 + n}{4} \quad (24)$$

$$= \frac{2n^3 + 3n^2 + n}{2} \quad (25)$$

- (c) Adding 1 because of Edstem, and assuming n is a power of 4 also according to Ed.

$$\sum_{i=1}^{\log_4 n^2 + 1} 1 \quad (26)$$

$$= \sum_{i=1}^{\log_4 n^2} 1 + 1 \quad (27)$$

$$= \log_4 n^2 + 1 \quad (28)$$

$$= 2 \log_4 n + 1 \quad (29)$$

(d)

$$\sum_{i=1}^n \sum_{j=1}^{\lfloor \log n \rfloor + 1} 1 \quad (30)$$

$$= \sum_{i=1}^n \lfloor \log n \rfloor + 1 \quad (31)$$

$$= n \lfloor \log n \rfloor + 1 \quad (32)$$

3 a

first insertions

0	
1	
2	
3	6003 → 6033 → 1953 → 2023 → 5763
4	
5	
6	
7	
8	3408 → 8458 → 2308 → 1028
9	

2 deletions

0	
1	
2	
3	6003 → 1953 → 2023
4	
5	
6	
7	
8	3408 → 8458 → 2308 → 1028
9	

6 probes

final insertions

0	
1	
2	
3	6003 → 1953 → 2023
4	7634
5	
6	
7	
8	3408 → 8458 → 2308 → 1028
9	4959

2 probes

3 b

insert 9

0	8458
1	3408
2	
3	5763
4	2023
5	1953
6	6033
7	6003
8	1028
9	2308

delete 2

0	8458
1	3408
2	
3	X
4	2023
5	1953
6	X
7	6003
8	1028
9	2308

5 probes

insert 2

0	8458
1	3408
2	4959
3	X
4	2023
5	1953
6	7634
7	6003
8	1028
9	2308

4 + 3 = 7 probes

3 c

first insertions

0	→ 8458
1	→ 3408
2	→ 2023
3	→ 6033 → 2308 → 1028
4	
5	→ 5763
6	
7	
8	→ 6003
9	→ 1953

2 deletions

0	→ 8458
1	→ 3408
2	→ 2023
3	→ 2308 → 1028
4	
5	
6	
7	
8	→ 6003
9	→ 1953

2 probes

8

final insertions

0	→ 8458
1	→ 3408
2	→ 2023
3	→ 2308 → 1028
4	
5	
6	→ 4959
7	
8	7634 → 6003
9	→ 1953

2 probes

(3d)

first insertions

0	8958
1	3408
2	2023
3	1028
4	2308
5	5763
6	6033
7	
8	6003
9	1953

2 deletion

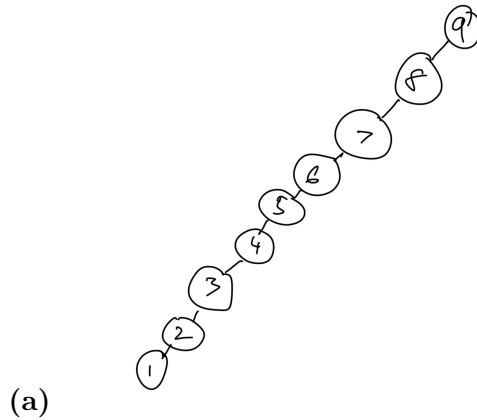
0	8958
1	3408
2	2023
3	1028
4	2308
5	X
6	X
7	
8	6003
9	1953

5 probes

2 insertion

0	8958
1	3408
2	2023
3	1028
4	2308
5	7634
6	4959
7	
8	6003
9	1953

9 probes

Problem 4.

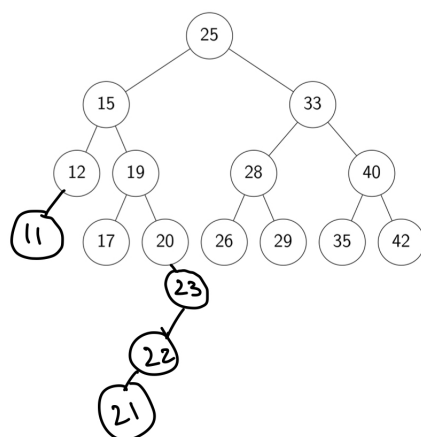
Assuming $n, n-1, n-2, \dots, 1$ are inserted into a BST, the runtime of this would be $O(\sum_{i=1}^n 1)$. This is because the first element inserted would be an $O(1)$ operation, then $O(2)$, and so on until $O(n)$. The resulting runtime is $O(\sum_{i=1}^n 1) = O(\frac{n(n+1)}{2}) = O(n^2)$.

To get big theta we need to prove that the runtime is also $\Omega(n^2)$. Ω for this would be the same, because there is no better way to insert the elements in descending order; we have to traverse the left child every time.

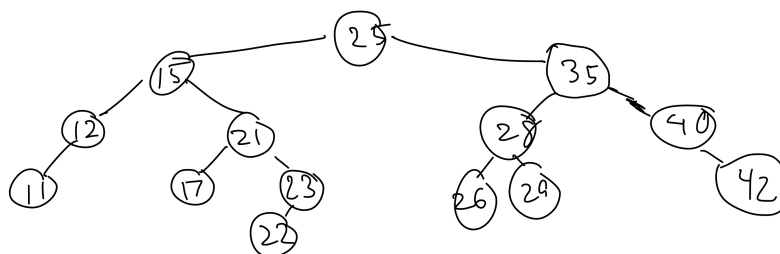
$$f(n) = O(n^2) \text{ and } g(n) = n^2$$

$$\text{When } C = 1, \text{ and } k = 1, n^2 \leq C \cdot n^2$$

Therefore, the runtime is $\Theta(n^2)$.



(b)

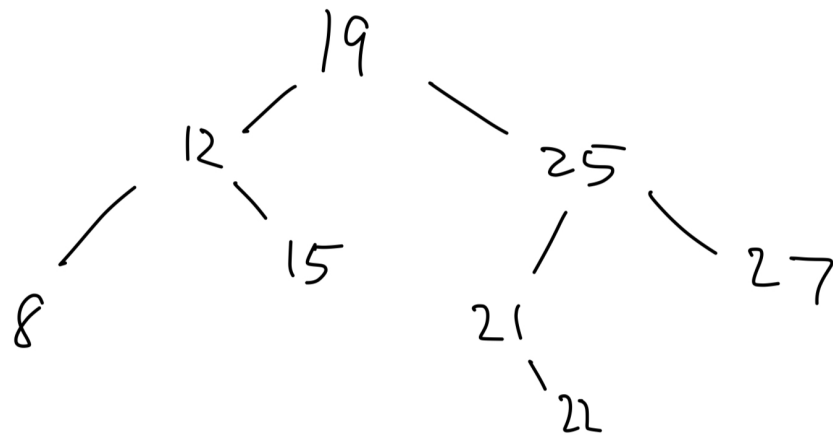


(c)

del 19: 19 had 2 children, so we replace 19 with $\text{min}(\text{right})$, which is 20. 20 has one child, so we replace 20 with 21. 21 has no children, so we delete it.

del 33: 33 had 2 children, so we replace 33 with $\text{min}(\text{right})$, which is 35. 35 has no children, so we delete it.

del 20: 20 has one child, so we take the min of the right subtree, which is 21, then delete 21.



(d)