
Problem Set 1

All parts are due February 18th, 2016 at 11:59PM.

Name: Alex List

Collaborators: /

Part A

Problem 1-1.

(a) $f_2 = \log(4n^{n^4}) = n^4 \log(n)$

$$f_2 > f_1 \rightarrow f_1 = O(f_2)$$

$$f_3 = 4\log(4n)\log(n) \simeq \log(n)\log(n) = \log^2(n)$$

$$\lim_{n \rightarrow \infty} f_3/f_2 = \lim_{n \rightarrow \infty} \log^2(n)/n^4 \log(n) = \lim_{n \rightarrow \infty} \log(n)/n^4 = 0$$

$$\rightarrow f_3 = O(f_2)$$

$$\lim_{n \rightarrow \infty} f_3/f_4 = \lim_{n \rightarrow \infty} \log^2(n)/\log^4(n) = 0 \rightarrow f_3 = O(f_4)$$

$$\lim_{n \rightarrow \infty} f_5/f_4 = \lim_{n \rightarrow \infty} \log^4(\log(n))/\log^4(n) = 0 \rightarrow f_5 = O(f_4)$$

$$\rightarrow f_1 < f_2 > f_3 < f_4 > f_5$$

Comparing f_1 & f_3

$$\lim_{n \rightarrow \infty} f_3/f_1 = \lim_{n \rightarrow \infty} \log^2(n)/n^4 = 0 \rightarrow f_3 = O(f_1)$$

$$\rightarrow f_3 < f_1 < f_2 \& f_5 < f_4 > f_3$$

Comparing f_2 & f_5

$$\lim_{n \rightarrow \infty} f_5/f_2 = \lim_{n \rightarrow \infty} \log^4(\log(n))/n^4 \log(n) = \lim_{n \rightarrow \infty} \log(\log(n))/n \log(n) =$$

$$0 \rightarrow f_5 = O(f_2)$$

$$\rightarrow f_3 < f_1 < f_2 \& f_2 > f_5 < f_4 > f_3$$

Comparing f_3 & f_5

$$\lim_{n \rightarrow \infty} f_5/f_3 = \lim_{n \rightarrow \infty} \log^4(\log(n))/\log^2(n) = \lim_{n \rightarrow \infty} \log(\log(n))/\log(n) =$$

$$0 \rightarrow f_5 = O(f_3)$$

$$\rightarrow f_5 < f_3 < f_1 < f_2 \& f_4 > f_3 \& f_4 > f_5$$

Comparing f_4 & f_1

$$\lim_{n \rightarrow \infty} f_4/f_1 = \lim_{n \rightarrow \infty} \log^4(n)/n^4 = 0 \rightarrow f_4 = O(f_1)$$

$$\rightarrow f_5 < f_3 < f_4 < f_1 < f_2$$

$$f_5, f_3, f_4, f_1, f_2$$

(b) Transform each function with logarithm and compare:

$$f'1 = \log(4^{4^n}) = 4^n \log(4)$$

$$f'2 = \log(4^{4^{n+1}}) = 4^{n+1} \log(4)$$

$$f'3 = \log(5^{4^n}) = 4^n \log(5)$$

$$f'4 = \log(5^{4n}) = 4n \log(5)$$

$$f'5 = \log(5^{5n}) = 5n \log(5)$$

By inspection:

$$f4 = O(f5), f1 = O(f2), f4 = O(f3), f1 = O(f3), f3 = O(f2), f5 = O(f3), f5 = O(f1), f4 = O(f1)$$

$$f4 < f5, f1 < f2, f1 < f3, f3 < f2, f4 < f3, f5 < f3, f5 < f1, f4 < f1$$

$$\rightarrow f4 < f5 < f1 < f3 < f2$$

$$f4, f5, f1, f3, f2$$

(c) Simply first then compare

$$f1 = \frac{n!}{4!(n-4)!} = O(n^4)$$

$$*f2 = \frac{n!}{(n/4)!(n-n/4)!} = \frac{n!}{(n/4)!(3n/4)!} = O(1) \text{ n ns in numerator, n ns in denominator.}$$

$$f3 = 4n! = O(n^n)$$

$$f4 = 4^{n/4}$$

$$f5 = (n/4)^{n/4}$$

Visually:

$$f2 < f1 < f3, f5 = O(f3), f1 = O(f5), f2 = O(f4)$$

$$\rightarrow f2 < f1 < f5 < f3, f2 = O(f4)$$

I'll compare f4 and f3. I wasn't sure, so I use L'Hospital's rule

$$\lim n \rightarrow \inf \frac{d}{dn} \frac{f4}{f3} = \lim n \rightarrow \inf \frac{d}{dn} \frac{4^{n/4}}{n^n} = \lim n \rightarrow \inf \frac{2^{n/2-1} \log(2)}{n^n (\log(n)+1)} = 0$$

$$\rightarrow f4 = O(f3)$$

I still need to compare f4 and f5. I'll use L'Hospital's rule again.

$$\lim n \rightarrow \inf \frac{d}{dn} \frac{f5}{f4} = \lim n \rightarrow \inf \frac{d}{dn} \frac{(n/4)^{n/4}}{4^{n/4}} = \lim n \rightarrow \inf \frac{2^{-n/2-2} n^{n/4} \log(n/4+1)}{2^{n/2-1} \log(2)} = 0$$

$$\rightarrow f5 = O(f4)$$

Because the $2^{-n/2-2}$ term goes to 0.

$$\rightarrow f2 < f1 < f5 < f4 < f3$$

$$f2, f1, f5, f4, f3$$

Problem 1-2.

(a) 1. $\theta(n)$ Branch factor 1. N iterations of c work.

2. $O(n^2)$ Because summation over work on all levels from c to nc is $\frac{n*(n+1)}{2} = \frac{n^2+n}{2} = O(n^2)$

3. $\theta(\log(n))$ Because $\log(n)$ levels with c work.
 4. * $O(n)$ Because with branching factor 2, work c per node, bottom row does most work, $c * 2^{\text{height}=\log(n)} = cn = O(n)$
 5. $\theta(n\log(n))$ Because there are $\log(n)$ rows of cn work per row.
 6. * $O(n^{\log(3)})$ Because with branching factor 3, the bottom row does most work $3^{\text{height}=\log(n)}$ nodes of c work $= c * n^{\log(3)/\log(2)} = O(n^{\log(3)})$
- (b)
1. $T(n) = T(n/2) + c$ Binary search has one subproblem, with half the nodes to visit. $O(\log(n))$ total.
 2. * What was name of the matrix algorithm? $T(n) = T(n/2) + c * \log(n)$ For $n \times n$ matrix, Instead of doing c work for 1 comparison, I search the n -width row, $\log(n)$ work per subproblem.

Problem 1-3.

- (a) The naive solution is $O(n^4)$ with 4-nested loops. It holds a variable named *maxGain* while (for abuy1 in A for first buy date (for asell1 > abuy1 in A for first sell date (for abuy2 >= asell1 in A for second buy date (for asell2 > abuy2 in A for second sell date (maxGain = A[asell1] - A[abuy1] + A[asell2] - A[abuy2] if A[asell1] - A[abuy1] + A[asell2] - A[abuy2] > maxGain))))
- code here code here
code here code here

(b)

Part B**Problem 1-4.**

Submit your implemented python script.

- (a)
- (b)
- (c)
- (d)
- (e)