Homework 03: Midterm Review

Due date: Monday, 7/23 at 11:59 pm

Instructions:

Submit a typed or neatly handwritten scan of your responses on Canvas in PDF format.

Note: you will need to submit a separate PDF per each section.

1. Big-O Notation

Let $f(n) = 10^6 \cdot n^2 + 10^8 \cdot n^{1.5} + n^3 + 10^{10} \cdot n^{2.99}$. Show that $f(n) = O(n^3)$ by finding a constant c and an integer n_0 and applying the Big-O definition.

(Next page)

2. Solving Recurrences

(a) For the following recurrence relations, state which case of the Master Theorem applies and give the Big-Θ runtime bound.

(i)

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ 7 \cdot T(n/2) + n^2 & \text{otherwise} \end{cases}$$

(Next page)

(ii)

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ 4 \cdot T(n/2) + n^2 & \text{otherwise} \end{cases}$$

(iii)

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ 2 \cdot T(n/2) + \sqrt{n} & \text{otherwise} \end{cases}$$

(iv)

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ 4 \cdot T(n/2) + n^3 & \text{otherwise} \end{cases}$$

(v)

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ 3 \cdot T(n/2) + n & \text{otherwise} \end{cases}$$

1 Big-O Notation

$f(n) = 10^6 \cdot n^3 + 10^8 \cdot n^{1.5} + n^3 + 10^{10} \cdot n^{2.99}$

$= 10^6 \cdot n^3 + 10^8 \cdot n^{\frac{3}{2}} + n^3 + 10^{10} \cdot n^{3-0.01} \le \underline{10^6} \cdot n^3 + \underline{10^8} \cdot n^3 + \underline{n^3} + \underline{10^{10}} \cdot n^3$

$\le \underbrace{(10^6 + 10^8 + 1 + 10^{10})}_{c} \cdot n^3$

$\therefore f(n) \le 10101000001 \cdot n^3$

when $n_0 = 1$ $c = 10101000001$

---

2. (i) $T(n) = \begin{cases} 1 & \text{if } n=1 \\ 7 \cdot T(n/2) + n^2 & \text{otherwise} \end{cases}$

Apply the master Theorem
let $a = 7$. $b = 2$. $c = 2$

if $\log_b a = \log_2 7 > 2$

Then $T(n)$ is $\boxed{\Theta(n^{\log_2 7})}$

(ii) $T(n) = \begin{cases} 1 & \text{if } n=1 \\ 4\,T(n/2) + n^2 & \text{otherwise} \end{cases}$

Apply the master theorem
let $a = 4$. $b = 2$. $c = 2$

if $\log_b a = \log_2 4 = 2$

Then $T(n)$ is $\boxed{\Theta(n^2 \log n)}$

(iii) $T(n) = \begin{cases} 1 & \text{if } n=1 \\ 2\,T(n/2) + \sqrt{n} = 2T(n/2) + n^{\frac{1}{2}} & \text{otherwise} \end{cases}$

Apply the master theorem. let $a = 2$ $b = 2$ $c = \frac{1}{2}$

if $\log_2 2 = 1 > \frac{1}{2}$ Then $T(n)$ is $\boxed{\Theta(n)}$

(iv) $T(n) = \begin{cases} 1 & \text{if } n=1 \\ 4T(n/2) + n^3 & \text{otherwise} \end{cases}$

Apply the master theorem
let $a = 4$, $b = 2$. $c = 3$

if $\log_2 4 = 2 < 3$ Then $T(n)$ is $\boxed{\Theta(n^3)}$

(v) $T(n) = \begin{cases} 1 & \text{if } n=1 \\ 3T(n/2) + n & \text{otherwise} \end{cases}$

Apply the master theorem
let $a = 3$. $b = 2$ $c = 1$

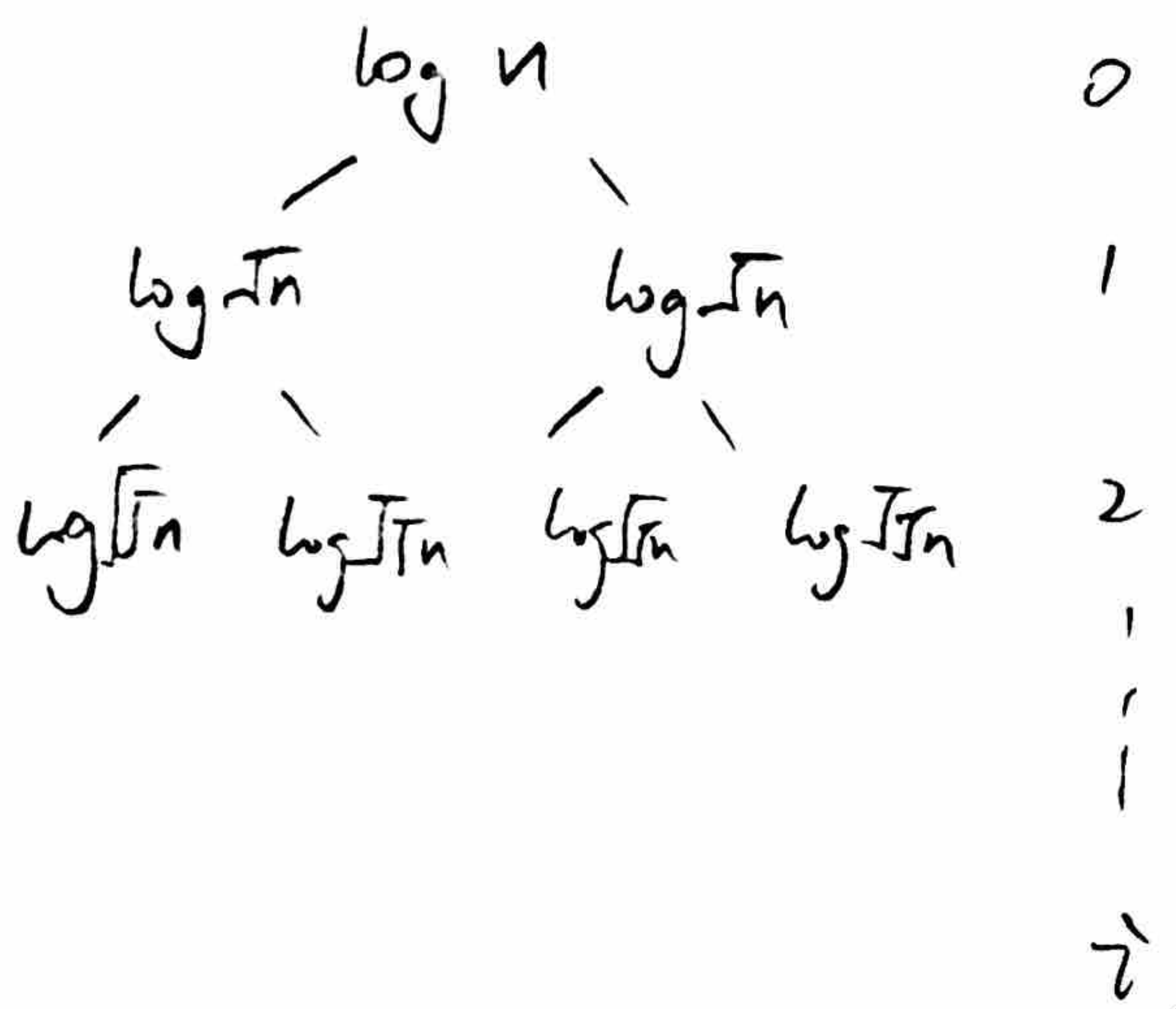
if $\log_2 3 > 1$ Then $T(n)$ is $\Theta(n^{\log_2 3})$

2 b) $T(n) = 2T(\sqrt{n}) + \log n$

$= 2(2T(\sqrt{\sqrt{n}}) + \log\sqrt{n}) + \log n$

$= 2(2(2T(\sqrt{\sqrt{\sqrt{n}}} + \log\sqrt{\sqrt{n}}) + \log n = \cdots$

Tree:



(others for next page!)

(b) Consider the recurrence

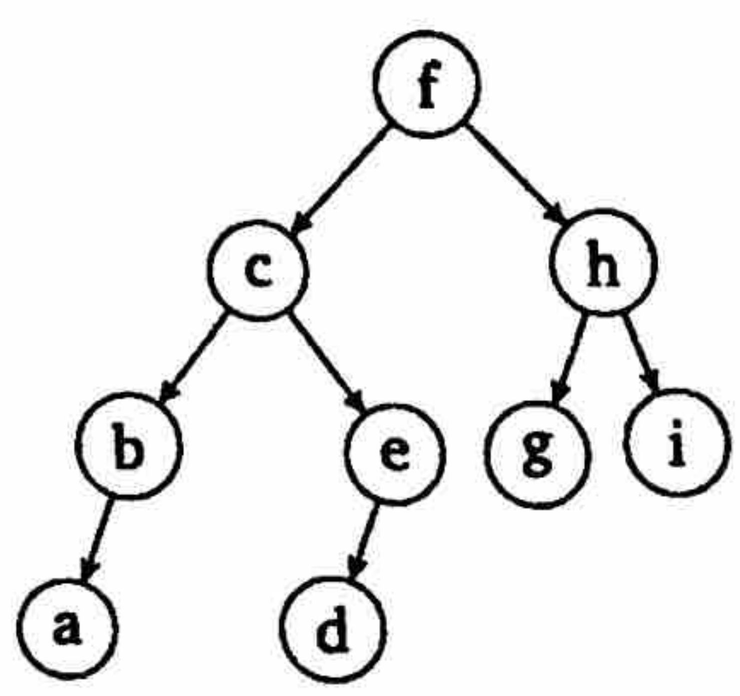
$$T(n) = \begin{cases} 1 & \text{if } n \leq 2 \\ 2 \cdot T(\sqrt{n}) + \log n & \text{otherwise} \end{cases}$$

Solve the above recurrence using the tree method. First unroll two levels of the tree and draw this unrolling as a tree. Finish your analysis by completing the missing entries of this table.

# of nodes at level i	2^i
input size at level i	$n^{\frac{1}{2^i}}$
work per node at level i	$\log n^{\frac{1}{2^i}} = \boxed{\frac{1}{2^i} \log n}$
total work at level i	$\cancel{2^i} \cdot \frac{1}{\cancel{2^i}} \log n = \boxed{\log n}$
level of base case	$n^{\frac{1}{2^i}} = 2 \Rightarrow \boxed{i = \log(\log n)}$
number of nodes in the base case level	$2^{\log(\log n)}$
expression for recursive work	$\sum_{i=0}^{\log(\log n)-1} \log n = \boxed{\log n * \log(\log n)}$
expression for non-recursive work	$1 * \log n = \boxed{\log n}$
closed form for total work	$\log n + \log n * \log(\log n)$
simplest Big-Θ for the total work	$\because \log n < \log n \cdot \log(\log n)$ $\therefore O(\log n \cdot \log(\log n))$

3. Binary Search Trees

(a) Consider the following Binary Search tree:



Write down the

(i) in-order traversal a b c d e f g h i