Question 1

$f(n) = n \lg n$ is asymptotically larger than $n^{\log_b a} = n$. The problem is that it is not *polynomially* larger. The ratio $f(n)/n^{\log_b a} = (n \lg n)/n = \lg n$ is asymptotically less than n^{ϵ} for any positive constant ϵ. Consequently, the recurrence falls into the gap between case 2 and case 3. (See Exercise 4.6-2 for a solution.)

4.6-2 ★

Show that if $f(n) = \Theta(n^{\log_b a} \lg^k n)$, where $k \geq 0$, then the master recurrence has solution $T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)$. For simplicity, confine your analysis to exact powers of b.

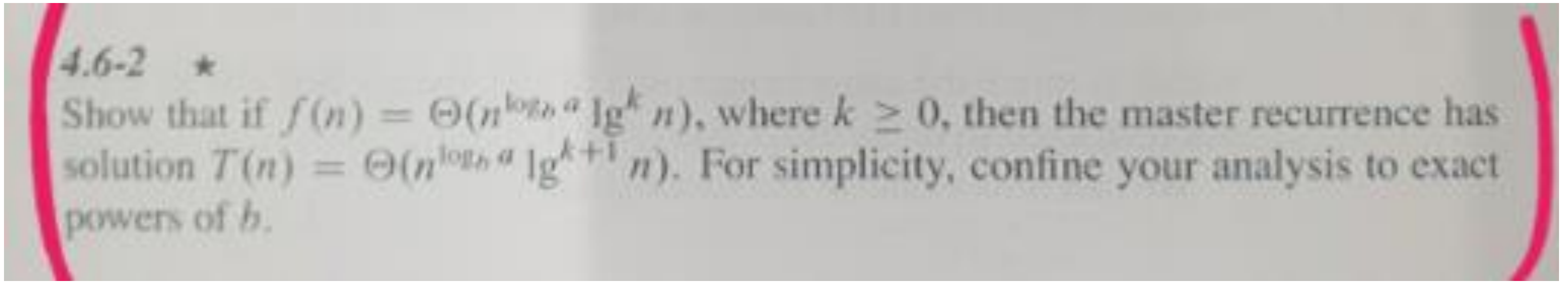
With master theorem

$$T(n) = 2T(n/2) + \Theta(n \lg(n))$$

a = 2, b = 2. Thus $n^{\log_b a} = n$

$f(n) = n\lg(n) = \Theta(n^{\log_b a} \lg^k n)$, where $k=1$.

Thus, $T(n) = \Theta(n^{\log_b a} \lg^{k+1} n) = \Theta(n \lg^2 n)$



4.6-2 ★
Show that if $f(n) = \Theta(n^{\log_b a} \lg^k n)$, where $k \geq 0$, then the master recurrence has solution $T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)$. For simplicity, confine your analysis to exact powers of b.

Proof

이 예제에 대한 풀이와 설명은 다음과 같습니다.

Proof. $T(n) = aT\left(\frac{n}{b}\right) + f(n)$

$\Rightarrow T(n) = \sum_{i=0}^{\log_b n} a^i f\left(\frac{n}{b^i}\right)$ — ① ($\because$ Recursive tree)

if $f(n) = \Theta(n^{\log_b a} \lg^k n)$ where $k \geq 0$,

①: $T(n) = \sum_{i=0}^{\log_b n} a^i \, \Theta\left(\left(\frac{n}{b^i}\right)^{\log_b a} \lg^k\left(\frac{n}{b^i}\right)\right)$

$= \Theta\left(\sum_{i=0}^{\log_b n} a^i \times \frac{n^{\log_b a}}{a^i} \lg^k\left(\frac{n}{b^i}\right)\right)$

$= \Theta\left(n^{\log_b a} \sum_{i=0}^{\log_b n} (\lg n - \lg b^i)^k\right)$ — ②

$(\lg n - \lg b^i)^k = \binom{k}{0}\lg^k n + \binom{k}{1}\lg^{k-1} n \lg b^i + \cdots + \binom{k}{k-1}\lg n \cdot \lg^{k-1} b^i + \binom{k}{k}\lg^k b^i$

$= \Theta(\lg^k n)$

②: $T(n) = \Theta\left(n^{\log_b a} \sum_{i=0}^{\log_b n} \Theta(\lg^k n)\right)$

$= \Theta\left(n^{\log_b a} \times (\log_b n + 1)\,\Theta(\lg^k n)\right)$

$= \Theta\left(n^{\log_b a} \times \Theta(\log_b n \cdot \lg^k n)\right)$

$= \Theta\left(n^{\log_b a} \cdot \Theta(\lg^{k+1} n)\right)$

$= \Theta(n^{\log_b a} \lg^{k+1} n)$

$\therefore$ if $f(n) = \Theta(n^{\log_b a} \lg^k n)$ where $k \geq 0$, $T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)$

알고리즘 책의 설명을 보아하니, 우리가 기존에 알고 있던 Master Theorem의 case 2

if $f(n) = \Theta(n^{\log_b a}) \Rightarrow T(n) = \Theta(n^{\log_b a} \lg n)$

이 공식은 위에서 증명된 결과에서 $k=0$일 때의 경우에 해당하는 특수한 케이스로,
polynomial difference를 만족하지 않아도 모든 $k \geq 0$에 대해 case 2를 적용할 수 있는
좀 더 일반화된 형태가

if $f(n) = \Theta(n^{\log_b a} \lg^k n) \Rightarrow T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)$

이 공식이라고 합니다

Binomial Coefficient

In mathematics, the **binomial coefficients** are the positive integers that occur as coefficients in the binomial theorem. Commonly, a binomial coefficient is indexed by a pair of integers $n \geq k \geq 0$ and is written $\binom{n}{k}$. It is the coefficient of the x^k term in the polynomial expansion of the binomial power $(1 + x)^n$

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}.$$

$$(1+x)^4 = \tbinom{4}{0}x^0 + \tbinom{4}{1}x^1 + \tbinom{4}{2}x^2 + \tbinom{4}{3}x^3 + \tbinom{4}{4}x^4$$
$$= 1 + 4x + 6x^2 + 4x^3 + x^4,$$

$$\begin{array}{ccccccccccc}
 & & & & & 1 & & & & & \\
 & & & & 1 & & 1 & & & & \\
 & & & 1 & & 2 & & 1 & & & \\
 & & 1 & & 3 & & 3 & & 1 & & \\
 & 1 & & 4 & & 6 & & 4 & & 1 & \\
1 & & 5 & & 10 & & 10 & & 5 & & 1
\end{array}$$

From wiki

위 종이에 적은 대로 저희가 기존에 알고 있는 case 2는 일반화된 case 2에서 k=0인 특수한 케이스로, 다시 말하면 위 식이 기존의 case 2를 확장한 형태라고 할 수 있습니다.

이전에 이 부분을 문의 드린 이유는, 교수님께서 LMS나 강의실에서 소개해주셨던 문제들 중에 case 2와 case 1, 3 사이의 gap으로 떨어지기에 Master Theorem으로 풀 수 없다고 소개해주셨던 문제들이 위 사실을 알고 나니 꼭 그렇지만도 않다는 것을 발견했기 때문이었습니다.

Show that if $f(n) = \Theta(n^{\log_b a} \lg^k n)$, where $k \geq 0$, then the master recurrence has solution $T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)$. For simplicity, confine your analysis to exact powers of b.

$$\begin{aligned}
g(n) &= \sum_{j=0}^{\log_b n - 1} a^j f(n/b^j) \\
f(n/b^j) &= \Theta\Big((n/b^j)^{\log_b a} \lg^k (n/b^j)\Big) \\
g(n) &= \Theta\Big(\sum_{j=0}^{\log_b n - 1} a^j \big(\frac{n}{b^j}\big)^{\log_b a} \lg^k \big(\frac{n}{b^j}\big)\Big) \\
&= \Theta(A) \\
A &= \sum_{j=0}^{\log_b n - 1} a^j \big(\frac{n}{b^j}\big)^{\log_b a} \lg^k \frac{n}{b^j} \\
&= n^{\log_b a} \sum_{j=0}^{\log_b n - 1} \Big(\frac{a}{b^{\log_b a}}\Big)^j \lg^k \frac{n}{b^j} \\
&= n^{\log_b a} \sum_{j=0}^{\log_b n - 1} \lg^k \frac{n}{b^j} \\
&= n^{\log_b a} B \\
\lg^k \frac{n}{d} &= (\lg n - \lg d)^k = \lg^k n + o(\lg^k n) \\
B &= \sum_{j=0}^{\log_b n - 1} \lg^k \frac{n}{b^j} \\
&= \sum_{j=0}^{\log_b n - 1} \Big(\lg^k n - o(\lg^k n)\Big) \\
&= \log_b n \lg^k n + \log_b n \cdot o(\lg^k n) \\
&= \Theta(\log_b n \lg^k n) \\
&= \Theta(\lg^{k+1} n) \\
g(n) &= \Theta(A) \\
&= \Theta(n^{\log_b a} B) \\
&= \Theta(n^{\log_b a} \lg^{k+1} n).
\end{aligned}$$

Chapter 16
Greedy Algorithms

Algorithm Analysis

School of CSEE

Exercise 1

Show counter example that following algorithm does not solve activity selection problem correctly.

The greedy approach of selecting the **activity of least duration** from among those that are compatible with previously selected activities.

The following example will pick a_2, but the solution is a_1, a_3.

i	1	2	3
s_i	0	2	5
f_i	5	6	10
Duration	5	4	5

Exercise 2

For the following set of frequencies:

a	b	c	d	e	f	g	h
1	1	2	3	5	8	13	21

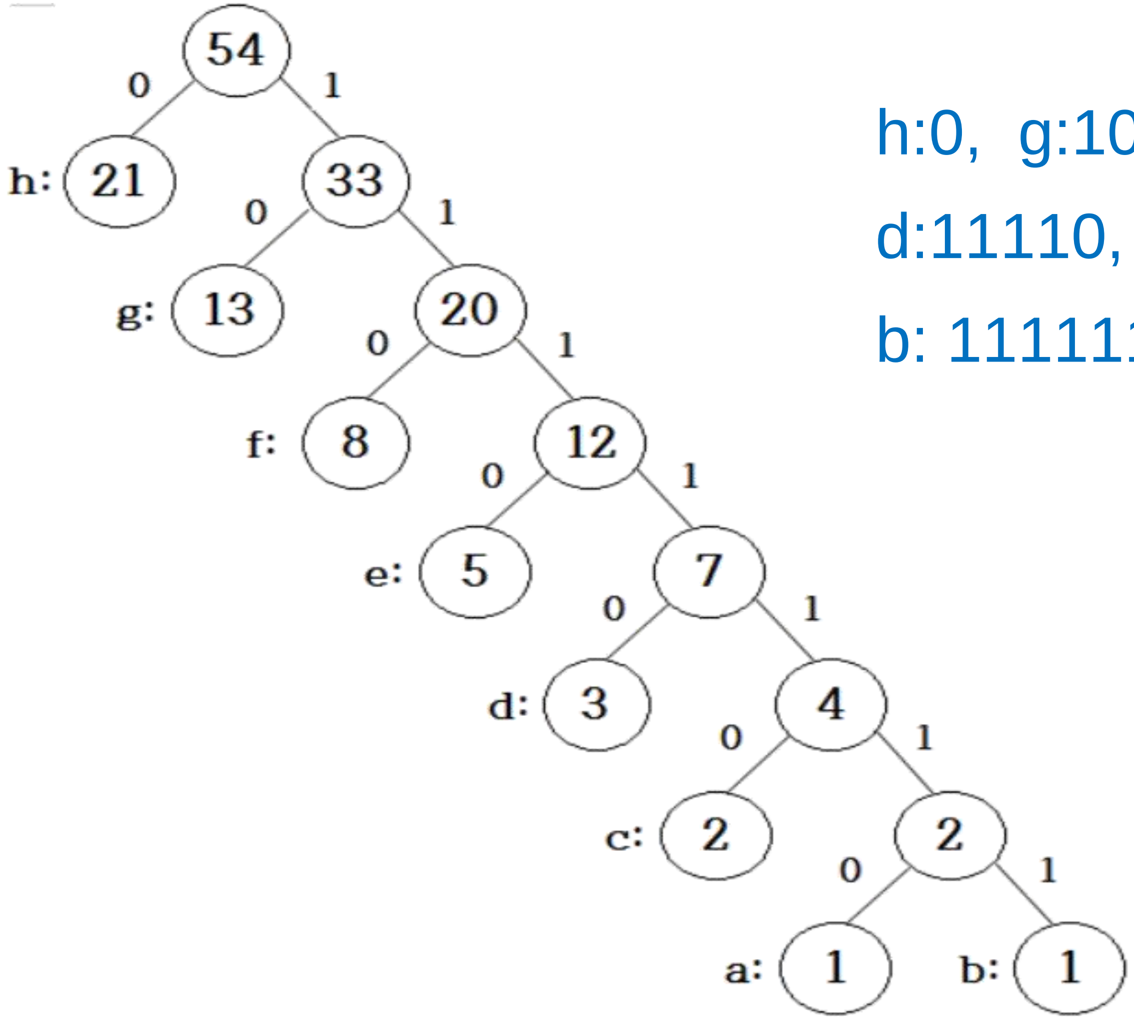
(a) After letter ‘c’ is extracted from queue (line 5 or 6), it will be combined with other node (line 7) and new node will be inserted to queue (line 8). Then what is the frequency of new node?

(b) What is an optimal Huffman code?

Left branch:0, right branch:1.

Answer 2

a:1, b:1, c:2, d:3, e:5, f:8, g:13, h:21



h:0, g:10, f:110, e:1110,

d:11110, c:111110,

b: 1111111, a:1111110