Homework 2 Solutions
Fundamental Algorithms, Fall 2004, Professor Yap

Due: Thu Oct 7, in class
SOLUTION PREPARED BY Instructor and T.A.s

INSTRUCTIONS:

- Please read questions carefully. When in doubt, please ask.

---

1. (20 Points)
Give the Θ-order for the following sums. If the sum is of polynomial-type or exponential-type sum, you must prove this fact before you invoke the general theorem for such sums. If it is neither polynomial-type nor exponential-type sum, you may be able to reduce to such a sum. [cf. Hw2 from Fall 2001]
(a) $S_n = \sum_{i=1}^n \binom{i}{3}$.
(b) $S_n = \sum_{i=1}^n (\lg^i n)$.
(c) $S_n = \sum_{i=1}^n \frac{i!}{\lg^i n}$.
(d) $S_n = \sum_{i=2}^n i^{1/\lg i}$.
Solution:
(a) Observe that $S_n = \frac{1}{6}\sum_{i=1}^n i^3 - 3i^2 + 2i$. Let's define $f(n) = \frac{1}{6}(n^4 - 3n^3 + 2n^2)$.
Note that by expanding $f(n) \leq c * f(n/2)$ we get $c = 32$ and $n > 4$ and therefore S_n is a polynomial-type sum and $S_n = \Theta(n^4)$.
Most students got it right.
(b) Observe $\lg^i n \geq 2\lg^{i-1} n$ for $i \geq 4$.
Hence S_n is a growing exponentially-type sum and $S_n = \Theta(\lg^n n)$.
Most students got it right.
(c) We divide this sum into 2 parts as follows: $S_n = \sum_{i=0}^{2\lg n} f(i,n) + \sum_{i=2\lg n+1}^{n} f(i,n)$. First, by using Stirling formula, which gives $n! = \theta((n/e)^{n+1/2})$, we get $\sum_{i=0}^{2\lg n} f(i,n) \leq 2\lg n f(2\lg n, n) = O(\frac{\lg n(\frac{2\lg n}{e})^{2\lg n+1/2}}{(\lg n)^{2\lg n}}) = O((\lg n)^{3/2})$. In order to show that $\sum_{i=2\lg n+1}^{n} f(i,n)$ is increasing exponentially we can take $c = 2$ when $i > 2\lg n$. Therefore, $S_n = \theta(\frac{n!}{\lg^n n})$ and by using Stirling formula it is equal to $\theta(\frac{(n/e)^n n^{1/2}}{\lg^n n})$.
Most students did not notice that it should be divided into 2 separate sums and/or did not expand $n!$ using Stirling formula. No points were taken off for those 2 issues.
(d) Note that $i^{1/\lg i} = 2$. Since $i \geq 2$ therefore $1/\lg i$ is defined. Hence $S_n = 2n$.
Several students did not notice that this is a sum over a constant.

2. (15 Points)
Use the Master Theorem to solve the following. You must justify why a given recurrence falls under any of the 3 possible cases.
(a) $T(n) = 18T(n/3) + n^3$.
(b) $T(n) = 27T(n/3) + n^3$.
(c) $T(n) = 18T(n/3) + n^2 \lg n$.
Solution:
(a) $a = 18$ and $b = 3$ therefore the watershed function is $W(n) = n^{\log_3 18}$ where $2 < \log_3 18 < 3$. This is case (+) because it is clear that $n^3 = \Omega(W(n)n^\varepsilon))$ for some $\varepsilon > 0$. In order to confirm the regularity condition we need to show that there there is a constant $c < 1$ such that $cn^3 \geq 18W(n/3)$. When solving $af(n/b) \leq cf(n)$ we find that $c \leq 2/3$. Hence, we conclude by the Master Theorem that $T(n) = \Theta(n^3)$.
Most students got it right.
(b) $a = 27$ and $b = 3$ therefore the watershed function is $W(n) = n^{\log_3 27} = n^3$. Hence, $f(n) = \Theta(w(n))$ and therefore case (0) applies. $T(n) = \Theta(w(n)\log n) = \Theta(n^3 \log n)$. Most students got it right.
(c) Same watershed function as in (a), however here case (-) applies. Therefore, $T(n) = \Theta(n^{\log_3 18})$.
Several students did not notice that this is case (-).

3. (15 Points)
Suppose algorithm A_1 has running time satisfying the recurrence

$$T_1(n) = aT(n/2) + n$$

and algorithm A_2 has running time satisfying the recurrence

$$T_2(n) = 2aT(n/4) + n.$$

Here, $a > 0$ is some constant. Compare these two running times for various values of a.

Solution:

The exponents of the watershed functions are, respectively, $\alpha_1 = \lg a$ and $\alpha_2 = (\lg a)/2 + (1/2)$, which are equal when $a = 2$.
When $a = 2$, $W_1(n) = W_2(n) = f_1(n) = f_2(n) = n$. Therefore this is case (0) and the solution for both equations after applying the Master theorem is $T(n) = \Theta(n \lg n)$.
When $a > 2$ then $\alpha_1 > \alpha_2$, $f_i(n) = W_i(n)n_i^{-\varepsilon}$ therefore this is case (-) and the solution is $\Theta(n^\alpha)$ for both cases.
When $a < 2$ then $\alpha_1 < \alpha_2$, $f_i(n) = n$ therefore this is case (+) and by applying the Master theorem we find that solution is $\Theta(n)$ for both cases.

Thus, we conclude that A_2 is never slower than A_1 in asymptotic running time.

4. (30 Points)
Use domain and range transformations to solve the following two recurrences:
(a) $T(n) = 4T(n/2) + n^2/\log n$.
(b) $T(n) = 4T(n/2) + n^2 \log n$.
Do not use real induction to solve this. You might wish to refer to a similar question in Hw2, Spring 2003.

Solution:
(a) By applying domain transformation of $n = 2^k$ we get $T(2^k) = 4T(2^{k-1}) + 4^k/k$. We then set $t(k)$ to be equal to $T(4^k)$ and get $t(k) = 4t(k-1) + 4^k/k$. By doing range transformation of dividing both sides by 4^k we define $S(k) = t(k)/4^k$ to get $S(k) = S(k-1) + 1/k$. By setting $S(0) = 0$ we get a telescopic series that sums up to $S(k) = \sum_{i=1}^{k} 1/k$. This is the Harmonic function, whose solution is $\ln k + \Theta(1)$. By applying the inverse of range transformation we get $t(k) = \Theta(4^k \ln k)$. By applying the inverse domain transformation we get $T(n) = \Theta(n^2 \ln \lg n)$.
(b) We apply the same domain transformation as in (a) and get $t(k) = 4t(k-1) + 4^k k$. To do range transformation we divide both sides by 4^k i.e., define $S(k) = t(k)/4^k$, to get $S(k) = S(k-1) + k$. After setting $S(0) = 0$ we get a telescopic series that sums up to $S(k) = \sum_{i=1}^{k} i$ and equals to $\Theta(k^2)$. Therefore, $t(k) = \Theta(4^k k^2)$ and $T(n) = \theta(n^2 \lg^2 n)$.

In both sections, several students did not correctly solve the arithmetic/harmonic series and/or did not perform the inverse transformations in order to get the final solutions for $T(n)$.

5. (20 Points)
Using real induction, give good upper and lower bounds for

$$T(n) = T(n - \lg n) + n.$$

HINT: first try to find an upper bound by expanding the recurrence and simplifying the intermediate expressions.

Solution:

The tightest result is $T(n) = \Theta(n^2/\lg n)$.

After expanding several steps of the recurrence we can see that

$$T(n) \geq T(n - i \lg n) + in - \sum_{j=1}^{i} i \lg n$$

This leads to $T(n) = \Omega(n^2/\lg n)$. Then by real induction we show that $T(n) = O(n^2/\lg n)$ as follows:

$$
\begin{aligned}
T(n) &= T(n - \lg n) + n \\
&\le C\frac{(n-\lg n)^2}{\lg(n-\lg n)} + n \\
&= C\frac{(n^2+\lg^2 n)}{\lg(n-\lg n)} - 2\frac{Cn\lg n}{\lg(n-\lg n)} + n \\
&\le C\frac{(n^2+\lg^2 n)}{\lg(n-\lg n)} - 2\frac{Cn\lg n}{\lg n} + n && \text{(since } \lg n \ge \lg(n-\lg n)\text{, we subtract a smaller quantity)} \\
&= C\frac{(n^2+\lg^2 n)}{\lg(n-\lg n)} - (2C-1)n && (*)
\end{aligned}
$$

Note that $\lg(n - \lg n) = \lg n + \lg(1 - \frac{\lg n}{n})$. Consider

$$
\begin{aligned}
\ln(1-x) &= -x - x^2/2 - x^3/3 - \dots && \text{(for } x < 1\text{)} \\
&= -x(1 + x/2 + x^2/3 + \dots) \\
&\ge -x(1 + x + x^2 + \dots) \\
&= \frac{-x}{1-x} \\
&\ge -2x && \text{(if } x \le \tfrac{1}{2}\text{).}
\end{aligned}
$$

Now $\lg(1-x) = \frac{1}{\ln 2}\ln(1-x) \ge \frac{-2x}{\ln 2}$. Using this we get $\lg(n-\lg n) \ge \lg n - \frac{2\lg n}{\ln 2n} = (1 - \frac{2}{\ln 2n})\lg n \ge (1 - \frac{3}{n})\lg n$. Continuing our argument from $(*)$ we have

$$
\begin{aligned}
T(n) &\le C\frac{(n^2+\lg^2 n)}{(1-\frac{3}{n})\lg n} - (2C-1)n \\
&= C\frac{(n^2+\lg^2 n)}{\lg n}\frac{n-3}{n} - (2C-1)n \\
&= C\frac{n^2}{\lg n}(1 + \frac{3}{n-3}) + \lg n(1 + \frac{3}{n-3}) - (2C-1)n \\
&\le C\frac{n^2}{\lg n} + \frac{16C(n-3)}{\lg n} + 4\lg n - (2C-1)n && \text{(since } 16(n-3)^2 \ge n^2 \text{ for } n \ge 4\text{)} \\
&\le C\frac{n^2}{\lg n} && \text{(for sufficiently large } C\text{).}
\end{aligned}
$$

The credits for this question are as follows: For only exploring the steps of the recurrence and not giving any bounds you got 5 points. For each reasonable bound and its proof you got 10 points. For a bound which was not reasonable but for which you gave a proof, you got 8 points. For only stating a correct bound without proving it you got 5 points. Only few students found the correct bounds and their proofs.