Homework 2
Fundamental Algorithms, Fall 2002, Professor Yap

Due: Thu Oct 3, in class

INSTRUCTIONS:

- TIP OF THE DAY: Your solution should be both LOGICAL (of course) and GRAMMATICAL (what?). Some students think that mathematical writing can ignore grammar, especially if you use some symbols. This is not true. Grammatical means it is constructed like a normal English writing: your solution is a sequence of complete ENGLISH sentences (even if your sentence involves mathematical symbols equations)! That means you begin each sentence with a capital letter, and end each sentence with a full stop. You may think the last point is trivial, until you try to grade a solution where you do not know where one sentence stops and the next begins!

---

1. (10 Points) The Fibonacci recurrence is

$$F(n) = F(n-1) + F(n-2).$$

(i) Suppose $\phi = \frac{1+\sqrt{5}}{2} = 1.61803...$ (this is also called the golden ratio) and $\widehat{\phi} = 1 - \phi = -0.61803....$ Show that ϕ and $\widehat{\phi}$ are solutions of the equation $x^2 = x + 1$
SOLUTION:

The roots of the equation $ax^2 + bx + c = 0$ can be obtained using the formula $x = \frac{-b \pm \sqrt{(b^2-4ac)}}{2a}$. Thus, the roots of the equation $x^2 - x - 1 = 0$ are given by $\frac{1 \pm \sqrt{1+4}}{2}$ i.e. ϕ and $\widehat{\phi}$.

(ii) Suppose the initial conditions are $F(0) = a$ and $F(1) = b$. Find α, β such that

$$F(n) = \alpha\phi^n + \beta\widehat{\phi}^n$$

is a solution.

SOLUTION:

We note that

- $F(0) = \alpha + \beta = a$.
- $F(1) = \alpha\phi + \beta\widehat{\phi} = b$.

We now have a pair of linear equations in the two variables α and β and they can be easily calculated to be $\alpha = \frac{b - a\widehat{\phi}}{\phi - \widehat{\phi}}$ and $\beta = \frac{a\phi - b}{\phi - \widehat{\phi}}$.

2. (20 Points) Consider the sum $T(n) = \sum_{i=1}^{n} f(i)$ in the following cases:
(i) $f(i) = i^3$
(ii) $f(i) = 1/i$
(iii) $f(i) = i/\log i$
(iv) $f(i) = i!$
(v) $f(i) = \frac{2^i}{i^2}$
(vi) $f(i) = (\log i)^{\log i}$
In each case, say if $T(n)$ is a polynomial-type sum, an exponential-type sum or neither. You must prove your claim. When it is polynomial-type or exponential-type sum, state the Θ-order of $T(n)$.

SOLUTION:

For every function we need to first check if it's decreasing or increasing. In case it's decreasing, we can eliminate the test for polynomial-type sums. Then, we can apply the tests required for exponential-type sums.

The tests are given here for reference

- **Polynomial type** $f(n)$ is increasing and $f(n) = O(f(\frac{n}{2}))$. In this case, $T(n) = \Theta(n.f(n))$.
- **Exponential type**

  (A) There exists a constant $C > 1$ such that for all i, $f(i) \geq C \cdot f(i-1)$. In this case, $T(n) = \Theta(f(n))$, or

  (B) There exists a constant $c < 1$ such that for all i, $f(i) \leq c \cdot f(i-1)$. In this case, $T(n) = \Theta(f(1))$.

(i) $f(i) = i^3$

SOLUTION:

We note that $f(i)$ is increasing with i and so we test for polynomial type.

We note that $i^3 \leq C \cdot (\frac{i}{2})^3$ for all $C \geq 8$. So, $f(i) = O(f(\frac{i}{2}))$. Hence, $T(n) = \Theta(n \cdot n^3) = \Theta(n^4)$.

(ii) $f(i) = \frac{1}{i}$.

SOLUTION:

We note that $f(i) = \frac{1}{i}$ is decreasing and hence is not a polynomial type sum. So, we test for the second case in exponential type sums i.e.,

$$\begin{aligned} \frac{1}{i} &\leq c_0 \cdot \frac{1}{i-1} \quad \text{for some } c_0 < 1 \\ \Rightarrow \frac{i-1}{i} &\leq c_0 \end{aligned}$$

But this condition doesn't hold for $i > i_0$ where $i_0 = \frac{1}{1-c_0}$. Thus, we conclude that $f(n)$ is neither polynomial-type nor exponential-type.

(iii) $f(i) = i/\log i$.

SOLUTION:

We note that $f(i)$ is increasing and so test for polynomial-type.

We have to show that $\frac{i}{\log i} \leq C \cdot \frac{\frac{i}{2}}{\log \frac{i}{2}}$ for some $C > 0$. We now have

$$\begin{aligned} \frac{\frac{i}{2}}{\log \frac{i}{2}} &= \frac{1}{2} \cdot \frac{i}{\log i - 1} \\ &> \frac{1}{2} \cdot \frac{i}{\log i} \end{aligned}$$

This implies that $\frac{i}{\log i} \leq 2 \cdot \frac{\frac{i}{2}}{\log \frac{i}{2}}$. Thus, $T(n) = \Theta(n \cdot \frac{n}{\log n}) = \Theta(\frac{n^2}{\log n})$.

(iv) $f(i) = i!$.

SOLUTION:

We note that $f(i)$ is increasing and guess that it's exponential-type. We proceed to make test (A) for exponential-type sums.

We need to show that $i! \geq C \cdot (i-1)!$ for some $C > 1$. But this holds for all $i > 2$ and hence $f(n)$ grows exponentially. Hence $T(n) = \Theta(n!)$.

(v) $f(i) = \frac{2^i}{i^2}$.

SOLUTION:

We note that $f(n)$ is increasing as $\frac{\frac{2^i}{i^2}}{\frac{2^{i-1}}{(i-1)^2}} = 2 \cdot (\frac{i-1}{i})^2 > 1$. We guess $f(i)$ grows exponentially large and make test (A).

We need to show that $\frac{2^i}{i^2} \geq C \cdot \frac{2^{i-1}}{(i-1)^2}$ for some $C > 1$ i.e., $2 \cdot (\frac{i-1}{i})^2 > C$. $\lim_{i \to \infty} \frac{i-1}{i} = 1$ and hence we have to choose $1 < C < 2$. Thus, $f(i)$ grows exponentially large and hence $T(n) = O(\frac{2^n}{n^2})$.

(vi) $f(i) = (\log i)^{\log i}$.

SOLUTION:

We note that $f(i)$ is increasing and show that it's neither polynomial-type nor exponential-type.

Assume that $f(i)$ is polynomial-type, i.e., there exists a $c > 0$ such that $(\log i)^{\log i} \leq c \cdot (\log \frac{i}{2})^{\log \frac{i}{2}}$ for i large enough. We will derive a contradiction. Taking logarithm of both sides, we get

$$\begin{aligned} & \log c + (\log(i/2) \log\log(i/2) &> \log i \cdot \log\log i \\ \Rightarrow \quad & \log c + (\log(i/2) \log\log i &> \log i \cdot \log\log i \\ \Rightarrow \quad & \log c &> \log\log i \cdot (\log i - \log(i/2)) \\ \Rightarrow \quad & \log c &> \log\log i \cdot \log 2 \end{aligned}$$

This is a contradiction since the R.H.S $\to \infty$ as $i \to \infty$. This proves that $f(i)$ is not polynomial-type.

Before we prove that $(\log i)^{\log i}$ is not exponential, we note a very useful fact, namely, $\log i - \log(i-1) = \Theta(\frac{1}{i})$. To see this, note that $\log i - \log(i-1) = \Theta(H(i) - H(i-1)) = \Theta(\frac{1}{i})$.

For exponential type, we need to show that there exists a $C > 1$ such that $(\log i)^{\log i} \geq C \cdot (\log(i-1))^{\log(i-1)}$. Taking logarithms of both sides, we get

$$\begin{array}{lcl} \log i \cdot \log\log i & \geq & \log C + \log(i-1)\cdot \log\log(i-1) \\ \log i \cdot \log\log i - \log(i-1)\cdot \log\log(i-1) & \geq & \log C \\ \log(i-1)(\log\log i - \log\log(i-1)) + \Theta(\frac{\log\log i}{i}) & \geq & \log C > 0 \end{array}$$

The term $\Theta(\frac{\log\log i}{i})$ arises from $(\log i - \log(i-1))\log\log i$. This term goes to 0 as $i \to \infty$. Next consider the term $\log(i-1)(\log\log i - \log\log(i-1))$. We have $\log\log i - \log\log(i-1) = \Theta(\int_{\log(i-1)}^{\log i} \frac{dx}{x}) = \Theta(\frac{\log i - \log(i-1)}{\log(i-1)})$ obtained by replacing $\frac{1}{x}$ with the largest value $\frac{1}{\log(i-1)}$ in the integral. So the term $\log(i-1)(\log\log i - \log\log(i-1)) = \log(i-1)\cdot\Theta(\frac{\log i - \log(i-1)}{\log(i-1)}) = \Theta(\frac{1}{i}) \to 0$ as $i \to \infty$. Thus, we have shown that the LHS goes to 0 but the RHS is positive (since $C > 1$). This is a contradiction.

In summary, we have shown that $f(i)$ is neither polynomial-type nor exponential-type.

3. Exercise 1.1 in the handout on Master Theorem. NOTE: no proofs are necessary for this question.

   State the solution, up to $\Theta-$ order of the following recurrences:

   (i) $T(n) = 10T(\frac{n}{10}) + \log^{10} n$.
   (ii) $T(n) = 100T(\frac{n}{10}) + n^{10}$.
   (iii) $T(n) = 10T(\frac{n}{100}) + (\log n)^{\log\log n}$.
   (iv) $T(n) = 16T(\frac{n}{4}) + 4^{\lg n}$.

   SOLUTION:

   We now state the Master theorem and then read off the $\Theta-$ order of the recurrences by stating the applicable case:

   THEOREM 1 *Let $a \geq 1$ and $b > 1$ be constants, let $f(n)$ be a function and let $T(n)$ be defined on the nonnegative integers by the recurrence*
   $T(n) = aT(\frac{n}{b}) + f(n)$.

   where we interpret $\frac{n}{b}$ to mean either $\lfloor \frac{n}{b} \rfloor$ or $\lceil \frac{n}{b} \rceil$. Then $T(n)$ can be bounded asymptotically as follows:

   (a) If $f(n) = O(n^{\log_b a - \epsilon})$ for some constant $\epsilon > 0$, then $T(n) = \Theta(n^{\log_b a})$.

   (b) If $f(n) = \Theta(n^{\log_b a})$, then $T(n) = \Theta(n^{\log_b a} \lg n)$.

   (c) If $f(n) = \Omega(n^{\log_b a + \epsilon})$ for some constant $\epsilon > 0$, and if $af(\frac{n}{b}) \leq cf(n)$ for some constant $c < 1$ and all sufficiently large n, then $T(n) = \Theta(f(n))$.

   (i) $T(n) = 10T(\frac{n}{10}) + \log^1 0n$.

   SOLUTION:

We note that $f(n) = \log^{10} n = O(n^{\log_{10} 100-\epsilon})$ and hence case (1) applies and so $T(n) = \Theta(n)$.

(ii) $T(n) = 100T(\frac{n}{10}) + n^{10}$.

SOLUTION:

We note that $f(n) = n^{10}$ and $n^{\log_b a} = n^2$. We see that $n^1 0 = \Omega(n^{2+\epsilon})$ and also, $100f(\frac{n}{10}) = \frac{n^{10}}{10^8} \leq \frac{1}{10^8} \cdot n^{10} = c \cdot f(n)$. Thus taking $c = \frac{1}{10^8} < 1$ works. Hence, we note that case 2 applies and $T(n) = \Theta(n^{10})$.

(iii) $T(n) = 10T(\frac{n}{100}) + (\log n)^{\log \log n}$.

SOLUTION:

Here, we need to compare $(\log n)^{\log \log n}$ and $n^{\frac{1}{2}}$. Thus, we have

$$\begin{array}{lll} & (\log n)^{\log\log n} & \leq \quad n^{\frac{1}{2}-\epsilon} \\ \iff & (\log\log n)^2 & \leq \quad (\frac{1}{2}-\epsilon)\cdot \log n \end{array}$$

which is true as $\log^k n = O(n^c)$ for all $k, c > 0$. Thus, case 1 applies and $T(n) = \Theta(n^{\frac{1}{2}})$.

(iv) $T(n) = 16T(\frac{n}{4}) + 4^{\lg n}$.

SOLUTION:

We have $f(n) = 4^{\lg n} = n^2 = n^{\log_4 16}$. So, case 2 appleis and we get $T(n) = \Theta(n^2 \log n)$.

4. Exercise 1.3 in the handout on Master Theorem. Solve the master recurrence when $f(n) = w(n) \log^k n$, for any k.

SOLUTION:

We have $T(n) = aT(\frac{n}{b}) + f(n)$. We make a domain transformation by substituting n with the form b^m and $t(m) = T(b^m)$. Then, we get $t(m) = at(m-1) + f(b^m)$. We now make a range transformation by dividing with a^m and setting $s(m) = \frac{t(m)}{a^m}$. We then obtain $s(m) = s(m-1) + \frac{f(b^m)}{a^m}$.

We set $s(0) = 0$ and obtain $s(m) = \sum_{i=1}^{m} \frac{w(b^i)\log^k b^i}{a^i}$ where $n = b^m$. Now,

$$\begin{array}{lll} \sum_{i=1}^{m} \frac{w(b^i)\log^k b^i}{a^i} & = & \sum_{i=1}^{m} \frac{a^i \log^k b^i}{a^i} \\ & = & \sum_{i=1}^{m} \log^k b^i \end{array}$$

First, we assume that $k \geq 0$. Then, we observe that this is a polynomial-type sum and hence
$\sum_{i=1}^{m} \log^k b^i = \Theta(l \cdot \log^k b^m) = \Theta(\log b^m \cdot \log^k b^m) = \Theta(\log^{k+1} n)$.

If $k = -1$, we have
$\sum_{i=1}^{m} \log^{-1}(b^i) = \log^{-1}(b) \cdot \sum_{i=1}^{m} \frac{1}{i} = \log^{-1}(b) \cdot \Theta(\log m) = \Theta(\log \log n)$

Finally, if $k < -1$, we have
$\sum_{i=1}^{m} \log^k b^i = \Theta \sum_{i=1}^{m} \frac{1}{i^k} = \Theta(1)$.

Using the fact that we note that $a^m = w(n)$, we have

$$T(n) = t(b^m) = s(m).(a^m) = \begin{cases} \Theta(w(n)\cdot \log^{k+1} n) & \text{if} \quad k \geq 0 \\ \Theta(w(n)\cdot \log\log n) & \text{if} \quad k = -1 \\ \Theta(w(n)) & \text{if} \quad k < -1 \end{cases}.$$

You should get most of the credit if you solved the case for $k \geq 0$ correctly.