Problem Set 5 Solutions

Read sections 1.7, 1.8, 5.1, 5.2, and 5.3 in Rosen.

Problem 1. **(15 points)** In 1997 there was a breakthrough on a long-standing problem in computer science called The Traveling Salesman Problem. The idea is that there is a salesman who wants to visit n cities scattered around the country. The problem is to determine the order in which he should visit the cities so as to minimize the total distance that he travels, assuming that he can go in a straight line between any two cities. Surprisingly, this problem is very difficult when the number of cities n is large.

The breakthrough was an algorithm that efficiently computes a nearly-optimal solution, a route for the salesman that has length at most 10% greater than the shortest possible route. The running time of the algorithm is described in the orginal publication as "nearly linear" in the number of cities. More precisely, the algorithm requires about $n \ln^{400} n$ steps.

(a) Prove that $n \ln^{400} n$ is $o(n^{1.1})$. This shows that the algorithm truly is "nearly linear" in an asymptotic sense. You may make use of L'Hopital's rule:

$$\text{If } \lim_{n\to\infty} f(n) = \infty \text{ and } \lim_{n\to\infty} g(n) = \infty, \text{ then } \lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{f'(n)}{g'(n)}.$$

Proof. We will first use L'Hopital's Rule to prove that the ratio of the two running times tends to zero.

$$\begin{aligned}
\lim_{n\to\infty} \frac{n \ln^{400} n}{n^{1.1}} &= \lim_{n\to\infty} \frac{\ln^{400} n}{n^{0.1}} \\
&= \left(\lim_{n\to\infty} \frac{\ln n}{n^{1/4000}} \right)^{400} \\
&= \left(\lim_{n\to\infty} \frac{1/n}{\frac{1}{4000} n^{-3999/4000}} \right)^{400} \\
&= \left(\lim_{n\to\infty} \frac{4000}{n^{1/4000}} \right)^{400} \\
&= 0^{400} \\
&= 0
\end{aligned}$$

The first step is simplification. The second step uses the fact that the limit of a product is the product of the limits. L'Hopital's rule is applied in the third step, and the remaining steps are simplifications.

The definition of a limit implies that for all $c > 0$, there exists an n_0 such that $n \ln^{400} n / n^{1.1} < c$ for all $n \geq n_0$. Rewriting this inequality as $n \ln^{400} n < cn^{1.1}$, we establish that $n \ln^{400} n = o(n^{1.1})$ by the definition of o. ∎

(b) Find a specific value of n such that $n^{1.1}$ is actually greater than $n \ln^{400} n$.

We forgot to rule out small values of n. Thus, for example, $n = 1$ is an acceptable answer worth full credit. However, you are a SUPERIOR HUMAN BEING if you followed the intent of the question and identified an n such that the inequality holds for that value and for all larger values.

Suppose that we choose $n = e^{50000}$, a number with more than twenty-one thousand digits. Then $n^{1.1}$ is equal to e^{55000}, and $n \ln^{400} n$ is equal to $e^{50000} \cdot 50000^{400} = e^{50000 + 400 \ln 50000} \approx e^{54328}$, which is a little smaller.

Note that setting n equal to e^{40000}— a paltry seventeen thousand digit number— would not suffice. Asymptotic notation can be misleading!

Problem 2. **(10 points)** Order the functions in each set below so that each function is either O or Θ of the next function. Indicate whether the stronger Θ relationship holds in each case. No proofs are required.

Example. n^2, n, $n^3/(n+1)$, $3n^2$

$$\begin{aligned} n &= O(n^2) \\ n^2 &= \Theta(3n^2) \\ 3n^2 &= \Theta(n^3/(n+1)) \end{aligned}$$

(a) $\ln n, \ln \sqrt{n}, \sqrt{\ln n}, n^{1/n}$

$$\begin{aligned} n^{1/n} &= O(\sqrt{\ln n}) \\ \sqrt{\ln n} &= O(\ln \sqrt{n}) \\ \ln \sqrt{n} &= \Theta(\ln n) \end{aligned}$$

(b) $(\sqrt{n})^{2 \ln \ln n}, n, n \ln n, \ln n^{\ln n}, \ln n!$

The notation $\ln n^{\ln n}$ was ambiguous, and so two solutions are acceptable.

$$\begin{aligned} n &= O(n \ln n) \\ n \ln n &= \Theta(\ln n!) \\ \ln n! &= O((\ln n)^{\ln n}) \\ (\ln n)^{\ln n} &= \Theta((\sqrt{n})^{2 \ln \ln n}) \end{aligned}$$

Alternatively,

$$\begin{aligned} \ln(n^{\ln n}) &= O(n) \\ n &= O(n \ln n) \\ n \ln n &= \Theta(\ln n!) \\ \ln n! &= O((\sqrt{n})^{2 \ln \ln n}) \end{aligned}$$

(c) $n!, n^n, (n/3)^n, n^{n+\ln \ln n}, e^{n \ln \ln n}$

$$\begin{aligned} e^{n \ln \ln n} &= O((n/3)^n) \\ (n/3)^n &= O(n!) \\ n! &= O(n^n) \\ n^n &= O(n^{n+\ln \ln n}) \end{aligned}$$

The hardest relation here was probably $(n/3)^n = O(n!)$. The integral method gives the following.

$$\begin{aligned} \ln n! &\geq \int_1^n \ln u \, du \\ &= n \ln n - n + 1 \\ n! &\geq e^{n \ln n - n + 1} \\ &\geq \left(\frac{n}{e}\right)^n \end{aligned}$$

The ratio $(n/3)^n$ divided by $(n/e)^n$ is equal to $(e/3)^n$, which tends to 0 as n goes to infinity. This implies an O-relationship rather than a Θ-relationship.

Problem 3. (10 points) Suppose that n ovals are drawn in a plane so that every pair of ovals intersects at either 0 or 2 points. Let x_n be the maximum number of regions that can be formed by such a configuration of ovals. We can express x_n using the recurrence below.

$$
\begin{aligned}
x_1 &= 2 \\
x_{n+1} &= x_n + 2n
\end{aligned}
$$

The region outside of any oval "counts"; for example, one oval defines two regions, because the oval has an inside region and an outside region.

(a) Solve this recurrence using the plug and chug method.

First, we repeatedly apply the recurrence equation.

$$
\begin{aligned}
x_n &= x_{n-1} + 2(n-1) \\
&= x_{n-2} + 2(n-2) + 2(n-1) \\
&= x_{n-3} + 2(n-3) + 2(n-2) + 2(n-1)
\end{aligned}
$$

We can now guess a general form and check it by applying the recurrence equation one more time.

$$
\begin{aligned}
x_n &= x_{n-i} + 2(n-i) + \ldots + 2(n-2) + 2(n-1) \\
x_n &= x_{n-i-1} + 2(n-i-1) + 2(n-i) + \ldots + 2(n-2) + 2(n-1) \\
&= x_{n-(i+1)} + 2(n-(i+1)) + 2(n-i) + \ldots + 2(n-2) + 2(n-1)
\end{aligned}
$$

Setting $i = n-1$, gives an expression for x_n.

$$
\begin{aligned}
x_n &= x_{n-i} + 2(n-i) + \ldots + 2(n-2) + 2(n-1) \\
&= x_1 + 2 + 4 + \ldots + 2(n-2) + 2(n-1) \\
&= 2 + 2 + 4 + \ldots + 2(n-2) + 2(n-1) \\
&= 2 + 2\sum_{j=1}^{n-1} j
\end{aligned}
$$

Now we get the closed form $x_n = n(n-1) + 2$ from the formula for the sum of an arithmetic series.

(b) Describe how to arrange n ovals so as to form x_n regions. No proof of correctness is required.

Make all the ovals circles of the same size. Place the second circle slightly right of the first, the third slightly right of the second, and so forth.

Problem 4. **(10 points)** Consider the following recurrence.

$$
\begin{aligned}
y_0 &= 1 \\
y_n &= \sum_{i=1}^{n} y_{n-i} \cdot 2^i
\end{aligned}
$$

(a) Guess the solution to the recurrence.

First, we compute the first few terms to form the basis for a guess.

$$
\begin{aligned}
y_0 &= 1 \\
y_1 &= 2 \cdot y_0 = 2 \cdot 1 = 2 \\
y_2 &= 2 \cdot y_1 + 4 \cdot y_0 = 2 \cdot 2 + 4 \cdot 1 = 8 \\
y_3 &= 2 \cdot y_2 + 4 \cdot y_1 + 8 \cdot y_0 = 2 \cdot 8 + 4 \cdot 2 + 8 \cdot 1 = 32 \\
y_4 &= 2 \cdot y_3 + 4 \cdot y_2 + 8 \cdot y_1 + 16 \cdot y_0 = 2 \cdot 32 + 4 \cdot 8 + 8 \cdot 2 + 16 \cdot 1 = 128
\end{aligned}
$$

The above calculation suggests the following claim.

Claim. The solution to the recurrence is $y_0 = 1$ and $y_n = 2^{2n-1}$ for $n \geq 1$.

(b) Prove your guess correct using strong induction.

Proof. The proof is by strong induction. Let $P(n)$ be the proposition that $y_0 = 1$ and $y_n = 2^{2n-1}$ for $n \geq 1$. In the base case, $P(0)$ holds trivially. In the inductive step, assume $P(0), \ldots, P(n)$ to prove $P(n+1)$. By the definition of y_{n+1}, we have:

$$
\begin{aligned}
y_{n+1} &= \sum_{i=1}^{n+1} y_{n+1-i} \cdot 2^i \\
&= y_0 \cdot 2^{n+1} + \sum_{i=1}^{n} y_{n+1-i} \cdot 2^i \\
&= 1 \cdot 2^{n+1} + \sum_{i=1}^{n} 2^{2(n+1-i)-1} \cdot 2^i \\
&= 2^{n+1} + 2^{2n+1} \cdot \sum_{i=1}^{n} \frac{1}{2^i} \\
&= 2^{n+1} + 2^{2n+1} \cdot \left(1 - \frac{1}{2^n}\right) \\
&= 2^{2(n+1)-1}
\end{aligned}
$$

In the second step, we split out the special term of the summation that contains y_0. The third step uses the induction hypotheses, and in the fourth step we pull a constant out of the summation. The fifth equation follows from the formula for the sum of a geometric series. The last step is simplication.

This shows that $P(0), \ldots, P(n)$ imply $P(n+1)$, and the claim is proved by strong induction. ■

Problem 5. **(10 points)** The following recurrence was invented by a late Italian renaissance scholar named Giardi Invertabinacci, who earned a certain repute in his age for mocking profound ideas, poetry, sculpture, and even architecture by constructing upside-down or backward replicas of the orignal. His scaled model of the famed Tower of Pisa still stands– upside down, but not leaning– outside Palermo. His one venture into mathematics was a recurrence relation, recorded in a letter to a contemporary mathematician and reproduced below in modern notation.[1]

$$\begin{aligned} x_1 &= 1 \\ x_2 &= 1 \\ x_{n+1} &= \frac{x_n x_{n-1}}{x_n + x_{n-1}} \end{aligned}$$

(a) Guess the solution to the recurrence. Your solution may be expressed in terms of Fibonacci numbers.

We compute the first few terms to form the basis for a guess at the general solution.

$$\begin{aligned} x_1 &= 1 \\ x_2 &= 1 \\ x_3 &= \frac{x_2 x_1}{x_2 + x_1} = \frac{1 \cdot 1}{1 + 1} = \frac{1}{2} \\ x_4 &= \frac{x_3 x_2}{x_3 + x_2} = \frac{\frac{1}{2} \cdot 1}{\frac{1}{2} + 1} = \frac{1}{3} \\ x_5 &= \frac{x_4 x_3}{x_4 + x_3} = \frac{\frac{1}{3} \cdot \frac{1}{2}}{\frac{1}{3} + \frac{1}{2}} = \frac{1}{5} \\ x_5 &= \frac{x_5 x_4}{x_5 + x_4} = \frac{\frac{1}{5} \cdot \frac{1}{3}}{\frac{1}{5} + \frac{1}{3}} = \frac{1}{8} \end{aligned}$$

[1] Okay, your TA just made all this up. But good story, eh?

Guess. The solution to the recurrence is $x_n = 1/F_n$.

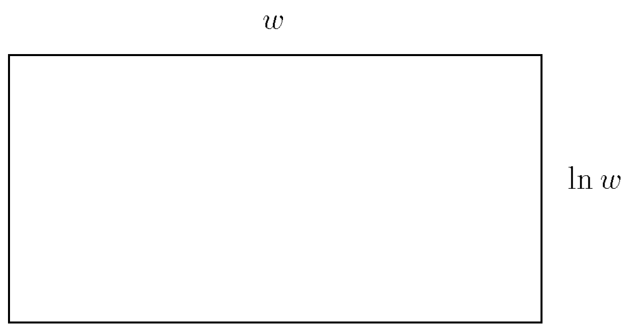
(b) Use strong induction to prove your guess correct.

Proof. The proof is by induction. Let $P(n)$ be the proposition that $x_n = 1/F_n$. In the base cases, $P(1)$ is true because $x_1 = 1 = 1/F_1$, and $P(2)$ is true because $x_2 = 1 = 1/F_2$. In the inductive step, for $n \geq 2$ assume $P(1), \ldots, P(n)$ to prove $P(n+1)$. From the defintion of x_{n+1}, we reason as follows:

$$\begin{aligned}
x_{n+1} &= \frac{x_n x_{n-1}}{x_n + x_{n-1}} \\
&= \frac{\frac{1}{F_n} \frac{1}{F_{n-1}}}{\frac{1}{F_n} + \frac{1}{F_{n-1}}} \\
&= \frac{1}{F_{n-1} + F_n} \\
&= \frac{1}{F_{n+1}}
\end{aligned}$$

The second step uses the induction hypotheses, the third step is simplification, and final step uses the Fibonacci recurrence. ■

Problem 6. **(20 points)** Recently, Theory Pig found a small piece of blue crayon behind the filing cabinet in Professor Lynch's office. Theory Pig was delighted to find that with this wondrous new toy he could draw pretty blue rectangles with a width of w centimeters and a height of $\ln w$ centimeters for any $w > 1$ that he took a fancy to.



Our pal proudly presented this remarkable discovery to 6.042 TA Rodrigo Leroux. Rodrigo was unimpressed: "Yeah, but what if I want a rectangle with an area of n square centimeters?

Then what should be the width w?" Fidgeting nervously with his crayon, Theory Pig replied that he believed the width should be $\Theta(n/\ln n)$ asymptotically, but that unfortunately he could not rigorously prove this fact. Rodrigo replied, "If you can't play with your toys properly, then they'll be taken away!" And then he confiscated Theory Pig's crayon!

Return joy to our little pal's world by supplying a proof of his claim.

Let $f(w) = w \ln w$. Then a rectangle with width w has area $f(w)$. Consequently, a rectangle with area n should have width $f^{-1}(n)$. We can now formalize the problem as follows.

Claim. If $f(w) = w \ln w$, then $f^{-1}(n) = \Theta(n/\ln n)$.

Proof. We must prove that there exists an n_0 and constants $c_1, c_2 > 0$ such that for all $n > n_0$

$$c_1 \cdot \frac{n}{\ln n} \geq f^{-1}(n) \geq c_2 \cdot \frac{n}{\ln n}.$$

Since f is an increasing function, the above inequalities hold if and only if

$$f\left(c_1 \cdot \frac{n}{\ln n}\right) \geq f(f^{-1}(n)) \geq f\left(c_2 \cdot \frac{n}{\ln n}\right).$$

Suppose that we choose $c_1 = e$ and $c_2 = 1$. Then the inqualities can be rewritten as follows:

$$e \cdot \frac{n}{\ln n} \cdot \ln\left(e \cdot \frac{n}{\ln n}\right) \geq n \geq \frac{n}{\ln n} \cdot \ln\left(\frac{n}{\ln n}\right)$$

$$n \cdot \left(e + \frac{e}{\ln n} - \frac{e \ln \ln n}{\ln n}\right) \geq n \geq n \cdot \left(1 - \frac{\ln \ln n}{\ln n}\right)$$

In the expression on the left, the $e \ln \ln n/\ln n$ term is $o(1)$, and so the expression in parentheses is greater than 1 for all sufficiently large n. In the expression on the right, the $\ln \ln n/\ln n$ term is positive for all $n > e$, and so the expression is parentheses is less than 1 for all sufficiently large n. Therefore, there exists an n_0 such that both inequalities hold for all $n > n_0$ as desired. ■

Problem 7. **(10 points)** We have devised a new version of `MergeSort`, which we call `MergeSortX19`, for the intellectually challenged inhabitants of planet Xoros 19.

Here is how the new algorithm works. The input is a list of n distinct numbers. If the list contains a single number, then there is nothing to do. If the list contains two numbers, then we sort them with a single comparison. If the list contains more than two numbers, then we perform the following sequence of steps.

- We make a list containing the first $\frac{2}{3}n$ numbers and sort it recursively.
- We make a list containing the last $\frac{2}{3}n$ numbers and sort it recursively.
- We make a list containing the first $\frac{1}{3}n$ numbers and the last $\frac{1}{3}n$ numbers and sort it recursively.
- We merge the first and second lists, throwing out duplicates.
- We merge this combined list with the third list, again throwing out duplicates.

The final, merged list is the output. We consider this algorithm ideal for the slow-witted creatures of Xoros 19, because multiple copies of each number are maintained. Therefore, even if the sorter occasionally forgets about a number, the algorithm can still output a complete, sorted list.

(a) Write a recurrence for the maximum number of comparisons needed to sort a list of n numbers using `MergeSortX19`. Assume that merging a list with l_1 numbers and a list with l_2 numbers and throwing out duplicates requires at most $2(l_1 + l_2)$ comparisons.

$$\begin{aligned} T(1) &= 0 \\ T(2) &= 1 \\ T(n) &= 3T\left(\frac{2}{3}n\right) + 2\left(\frac{2}{3}n + \frac{2}{3}n\right) + 2\left(n + \frac{2}{3}n\right) \\ &= 3T\left(\frac{2}{3}n\right) + 6n \end{aligned}$$

(b) Give a Θ bound on the maximum number of comparisons used by `MergeSortX19` to sort a list of n numbers.

We can apply the Akra-Bazzi method. First, we find p as shown below.

$$\begin{aligned} 3 \cdot \left(\frac{2}{3}\right)^p &= 1 \\ \ln 3 - p \ln \frac{3}{2} &= 0 \\ p &= \frac{\ln 3}{\ln \frac{3}{2}} \end{aligned}$$

Now the Θ bound on the maximum number of comparisons follows from an integration.

$$\begin{aligned} x_n &= \Theta\left(n^p \cdot \left(1 + \int_1^n \frac{6u}{u^{1+p}}\, du\right)\right) \\ &= \Theta\left(n^p\right) \\ &= \Theta\left(n^{2.7095\ldots}\right) \end{aligned}$$

Problem 8. **(15 points)** Use the Akra-Bazzi method to find a Θ bound on the solution to each of the following recurrences.

(a) $T(n) = 4T(\lfloor \frac{n}{3} \rfloor) + n^2$

$$4 \cdot (1/3)^p = 1 \quad \Rightarrow \quad p = \ln_3 4$$

$$\begin{aligned} T(n) &= \Theta\left(n^p \cdot \left(1 + \int_1^n \frac{u^2}{u^{p+1}}\, du\right)\right) \\ &= \Theta(n^2) \end{aligned}$$

(b) $T(n) = T(\lfloor \frac{3n}{5} \rfloor) + T(\lfloor \frac{4n}{5} \rfloor) + n^2$

$$(3/5)^p + (4/5)^p = 1 \quad \Rightarrow \quad p = 2$$

$$\begin{aligned} T(n) &= \Theta\left(n^2 \cdot \left(1 + \int_1^n \frac{u^2}{u^{2+1}}\, du\right)\right) \\ &= \Theta(n^2 \ln n) \end{aligned}$$

(c) $T(n) = T(\lfloor \frac{n}{3} \rfloor) + 2T(\lfloor \frac{n}{4} \rfloor) + n\sqrt{n}$

$$(1/3)^p + 2(1/4)^p = 1 \quad \Rightarrow \quad p < 1$$

$$\begin{aligned} T(n) &= \Theta\left(n^p \cdot \left(1 + \int_1^n \frac{u^{3/2}}{u^{p+1}}\, du\right)\right) \\ &= \Theta\left(n^p \cdot \left(1 + \int_1^n u^{1/2-p}\, du\right)\right) \\ &= \Theta(n\sqrt{n}) \end{aligned}$$