22C:44 Homework 3

Due by 5 pm on Thursday, 2/22

---

1. Attempt to solve the following recurrences using the Master Theorem. If you are unable to solve a recurrence using the Master Theorem, explain precisely why your attempt failed. For each recurrence, assume that it holds for all $n > n_0$ for some unspecified positive constant n_0. Further assume that $T(n) = \Theta(1)$ for $n \leq n_0$.
   (a) $T(n) = 5T(n/2) + n^2$.
   (b) $T(n) = 5T(n/2) + n\sqrt{n}$.
   (c) $T(n) = aT(n/a) + n^2 \lg n$.
2. Solve the following recurrences using the substitution method ("guess and verify"). Assume that each recurrence holds for $n > 1$ and that $T(n) = \Theta(1)$ for $n \leq 1$ and
   (a) $T(n) = T(n/2) + T(n/3) + n$. Guess: $T(n) = O(n)$.
   (b) $T(n) = T(n/3) + T(2n/3) + n$. Guess: $T(n) = \Omega(n \lg n)$.
   (c) $T(n) = T(\alpha n) + n$, where $0 < \alpha < 1$. Guess: $T(n) = \Theta(n)$.
3. Problem 4-2 (on Page 73).
   To show that the running time of the algorithm is $O(n)$, first set up a recurrence relation for the running time and solve it using your favorite method.
   Hint: If every one of the numbers were present, there would be $\lceil n/2 \rceil$ even numbers and $\lfloor n/2 \rfloor$ odd numbers. One scan of n bits tells you whether the missing number is odd or even. The algorithm can then proceed to examine only the odd numbers or only the even numbers.
4. Problems 7.2-1 (Page 144) and 7.3-1 (Page 147).
