Q1. Problem 4-1 parts a through h, and Problem 4-4 parts i and j only in CLRS (Recurrence examples and More recurrence examples). You need only give tight upper bounds (you may wish to do the lower bounds as an additional exercise, but only upper bounds will be graded). If you cannot get a tight upper bound on the recurrences, give the best upper bound you can get. You must justify your answers. That means either (1) solving the recurrence explicitly via telescoping and transformations, (2) guessing a bound and proving it by induction, (3) applying the Master Theorem, or (4) using recursion trees. Specify which of these techniques you used, and show as much of your work as you can.

Q2. Problem 4-2 in CLRS (Find the missing integer).

Q3. Exercise 5.4-2 in CLRS (Balls into bins).

Q4. Problem 5-2 in CLRS (Searching an unsorted array).

Q5. Problem 6-1 in CLRS (Building a heap using insertion).

Q6. Problem 7-4 in CLRS (Stack depth for quicksort).

Q7. Consider the Randomized-Quicksort algorithm (presented at the end of Section 7.3 of CLRS). Suppose that we view random bits as a scarce resource, and thus we will charge $\log_2 (r - p + 1)$ “random units” for generating a random integer in the range $p . . . r$ (as is done in Step 1 of Randomized-Partition algorithm). (Intuitively, this is because it takes $\log_2 m$ random bits to choose a random integer in the range $1 . . . m$.)

   (a) Show that in the worst case, the total number of “random units” used by Randomized-Quicksort is $\Theta(n \log n)$.

   (b) Show that the expected number of “random units” used is $O(n)$. Hint: The following inequality may be useful: $\log_2 n \leq \log_2 (n - 1) + 2/(n \ln(2))$ for $n \geq 2$.

Q8. Problem 8-4 in CLRS (Water jugs).