Homework #4: Binomial and Fibonacci Heaps, Disjoint Set Operations
Due Date: Tuesday, March 3, 1998 (9:00am)

This week's homework will ask you to simulate, by hand, many of the data structure operations which we have been covering in class. *You must work entirely on your own this week* . Due to the nature of the problems, we will not be giving any practice exercises this week, however the textbook works through several examples of operations during the chapters, and you should consider understanding those examples as perfect practice for this homework.

## Reading

Read carefully Ch. 20, Ch. 21, Ch. 22 of CLR.

## Problems

1. a. Exercise 20.2-3 of CLR, (10 points).
   b. Exercise 20.2-4 of CLR, (20 points).

   *Note: During the execution of* `Union`*, it may be that the working list contains three consecutive trees of equal degree. The final outcome depends on which two of the three trees you link in this situation. We will gladly accept any such outcome; you need not worry about making sure that you choose the identical two trees as the specific code in the text.*

2. a. Exercise 21.2-1 of CLR, (20 points).
   b. Exercise 21.4-1 of CLR, (20 points).

   *Note: Again, there is some non-determinism throughout the operations, which depends on the exact order of the circular lists. We will accept any valid outcome, so long as it could result from the correct execution of the operation in question.*

3. a. Exercise 22.2-2 of CLR, (10 points).
   b. Exercise 22.3-1 of CLR, (20 points).

## Extra Credit

(Exercise 22.3-4 of CLR.)

Show that any sequence of $m$ `Make-Set`, `Find-Set`, and `Union` operation, where all the `Union` operations appear before any of the `Find-Set` operations, takes only $O(m)$ time if both path compression and union by rank are used. What happens in the same situation if only the path-compression heuristic is used?