

Lab Exercises. (Due: Tues., 30 October 2001 in lab)

Copy the directory `$master/hw/lab7` to a directory of your own, using commands such as

```
cp -r $master/hw/lab7 .
```

In this directory, you should find a file called `README.lab7`, with directions about what to do during the scheduled lab. We strongly suggest that you read over this file *before* going to your lab.

We intend that you finish the lab exercises *in lab* and have your TA check them off. You can have any TA in any lab section check off your lab.

Homework Exercises. (Due: Fri., 26 October 2001 at midnight) Create a directory to hold your answers to this homework set. Copy the files from `$master/hw/hw7` into this directory. Put non-program answers into a file `hw7`. Use the command `submit hw7` to submit your solutions to the problems below.

1. Fill in a concrete implementation for the type `QuadTree` that has the following constructor:

```
/** An initially empty quadtree that is restricted to contain points
 * within the W x H rectangle whose center is at (X0,Y0). */
public QuadTree (double x0, double y0, double w, double h) ...
```

and no other constructors. See `~cs61b/hw/hw7/QuadTree.java`.

2. Assume that we have a heap that is stored with the largest element at the root. To print all elements of this heap that are greater than or equal to some key X , we *could* perform the `removeFirst` operation repeatedly until we get something less than X , but this would presumably take worst-case time $\Theta(k \lg N)$, where N is the number of items in the heap and k is the number of items greater than or equal to X . Furthermore, of course, it changes the heap. Show how to perform this operation in $\Theta(k)$ time *without* modifying the heap. See `~cs61b/hw/hw7/HeapStuff.java`