COP 3502 Suggested Program Edits/Questions: Heaps/Hash Tables (Week 12 Programs)

1) Take the heap sort in the posted file heap.c and compare its run time against implementations of merge and quick sort for up to 10,000,000 numbers.

2) The posted insert function in heap.c doesn't return a value in all cases (a bug). Fix the function and then test it to the point where the return value from the function matters.

3) Load various dictionaries into hashtable.c and for each calculate the length of the longest linked list in the table. Play around with the hash function to see if you can reduce this value via a change to the hash function.

4) Adjust htablelinear.c so that you read in a dictionary and make the table size about twice the size of the actual table. Then, calculate the average # of slots we would have to look at to insert a random value. (To do this, pretend that each hash value was equally likely and add up how many slots you have to iterate through before you hit an empty slot, including the empty slot, for each possible index in the array and then divide by the size of the array.)

5) Set up a test between htablelinear.c and hashquadratic.c to see which performs better! This is open ended so that you think about how to set up the test on your own in a fair way. (Adjust the code as necessary to make the comparison a meaningful one.)