The rows of your table will consist of the following names of sorting algorithms:

* Insertion
* Bubble (yeah, I know I said you shouldn't ever implement this one!)
* Selection
* Merge
* Quick

and the columns should be the same as those in either book. For each

*i*=1,2,3,4,5,6

, you are to take an array of size

10*i*

filled, in order, with the numbers 0 up to

10*i*−1

, randomly shuffle it, and then sort it using each of the above algorithms, counting the number of swaps and comparisons that each performs. (In the merge sort, each assignment to \*A[curr] is 1 swap, and each test \*temp[i1] <= \*temp[i2] is 1 comparison. In QuickSort, each comparison to the pivot is 1 comparison.) Please note that you are to create one random array of each size and then sort the same array using each of the algorithms, so you'll need to keep a copy. To perform the random shuffle, use this algorithm (known as the [Fisher-Yates shuffle](http://en.wikipedia.org/wiki/Fisher%E2%80%93Yates_shuffle)):

To shuffle an array a of n elements (indices 0..n-1):

for i from n − 1 downto 1 do

j ← random integer with 0 ≤ j ≤ i

exchange a[j] and a[i]

As usual, make sure the random-number generator is seeded before using it. For the last two columns of the table, you are using a 10,000-element unshuffled array containing, for Up, the numbers in increasing order and, for Down, the numbers in decreasing order.

Everything should be done with just one file, called sorting.cpp, that runs without input, and outputs the table in this format:

Algorithm 10 100 1,000 10,000 100,000 1,000,000 10K Up 10K Down

--------- --------- --------- --------- --------- --------- --------- --------- ---------

Insertion swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp

Bubble swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp

Selection swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp

Merge swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp

Quick swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp swap,comp

You may need to adjust the column widths depending on the sizes of the numbers in each column. (The commas in each column do not need to line up, but you can line them up if you like.)