Overview
This homework contains three problems whose solutions must be turned in at the start of class on Tuesday, October 30th. One of the TAs will pick up the solutions within the first 10 minutes of class. After that time no homework will be accepted. Solutions will be posted that afternoon and graded papers will be returned Friday, November 2nd.

This homework also contains a number of additional suggested problems to use as extra practice.

Graded Problems
1. (15 points)
   (a) Rewrite the MergeArrays function so that it does all of the work inside the single while loop, eliminating the two for loops at the end. Perhaps surprisingly, this will only involve minor changes to the while loop condition and the two if conditions.
   (b) Starting from this rewrite, rewrite MergeArrays again to merge three sorted arrays simultaneously.
2. (15 points) Carrano & Prichard, page 321, number 6. You only need to write a main program, similar to the one given in class.
3. (15 points) How can you use one or more stacks to simulate a queue? Answer by outlining how the enqueue, dequeue, and getFront operations of the queue must occur. You may only use the public stack and queue interfaces.

Extra Suggested Problems
1. Hand-simulate MergeSort on a small array. Add selected cout statements to the code provided as part of the lecture (it is available through the web page) to ensure that your hand simulation is correct.
2. Rewrite BinarySearch to make it non-recursive.


4. Suppose a queue and a stack each have a unique set of integers. Can you use the public interface to a queue to count the number of items stored and leave the queue unchanged? Can you do the same thing for a stack? For each, answer why or why not?