**Lab 16 Implementing bubble sort**

In this lab, you will implement the bubble sort algorithm. The bubble sort is so called because it compares adjacent items, "bubbling" the smaller one up toward the beginning of the array. By comparing all pairs of adjacent items starting at the end of the array, the smallest item is guaranteed to reach the beginning of the array at the end of the first pass.  
  
The second pass begins again at the end of the array, ultimately placing the second smallest item in the second position. During the second pass, there is no need to compare the first and second items, because the smallest element is guaranteed to be in the first position.  
  
Bubble sort takes at most n - 1 passes for an array of n items. During the first pass, n - 1 pairs need to be compared. During the second pass, n - 2 pairs need to be compared. During the ith pass, n - i pairs need to be compared. During the last pass, n - (n - 1) or one pair needs to be compared. If, during any pass, no two adjacent items need to be interchanged, the array is in order and the sort can terminate. If it continues, no further interchanges will occur.  
  
Write a tester program to test the correctness of your bubble sort.