計算機概論 作業 5

Bubble Sort (Exercise 6.11)

The bubble sort presented in Fig. 6.15 is inefficient for large arrays. Make the following simple modification to improve the performance of the bubble sort.

a) After the first pass the largest number is guaranteed to be in the highest-numbered element of the array; after the second pass, the two highest numbers are “in place,” and so on. Instead of making nine comparisons on every pass, modify the bubble sort to make eight comparisons on the second pass, seven on the third pass and so on.

b) The data in the array may already be in the proper order or near-proper order, so why make nine passes if fewer will suffice? Modify the sort to check at the end of each pass whether any swaps have been made. If none has been made, then the data must already be in the proper order, so the program should terminate. If swaps have been made, then at least one more pass is needed.