Andrew Taylor

Atayl136

Homework 8: Sorting

1. How many comparisons and interchanges (in terms of file size n) are performed by Simple insertion sort for the following files:

i. A sorted file

In each pass of an insertion sort comparisons occur, except for the first pass. So the total number of comparisons is n-1 . no interchanges take place because it is already sorted.

ii. A file that is sorted in reverse order (that is, from largest to smallest)

It is $\frac{n(n-1)}{2}$ comparisons and interchanges.

iii. A file in which x[0], x[2], x[4]… are the smallest elements in sorted order, and in which x[1], x[3], x[5]… are the largest elements in sorted order, e.g. [3, 14, 5, 15, 9, 8, 11, 19].

It is 2(n-1) comparisons and n+1.

2. How many comparisons and interchanges (in terms of file size n) are performed by Shell Sort using increments 2 and 1 for the following files:

i. A sorted file

The shell sort is an insertion sort with sublists divided from a gap factor. Sublists are separately sorted using insertion sort and then combined into one list which is insertion sorted a final time. So

$\frac{n-1}{2} + \frac{n-1}{2} + \frac{n}{2} = \frac{3n}{2} - 1$

No interchanges occur because it is sorted.

ii. A file that is sorted in reverse order (that is, from largest to smallest)

In the worst case shell sort is O(N^2). So it is K*N^2 comparisons.

iii. A file in which x[0], x[2], x[4]… are the smallest elements in sorted order, and in which x[1], x[3], x[5]… are the largest elements in sorted order, e.g. [3, 14, 5 15, 9, 18, 11, 19].

In this case it depends on the shell sort gap factor. If the gap factor is 2, the list is perfectly split into two sorted sublists, which is then combined into one sorted list. Shell sort still has combine these lists so it interchanges n times. The gap reduces to 1 on the last pass.

3. Determine the number of comparisons (as a function of n and m) that are performed in merging two ordered files a and b of sizes n and m, respectively, by the merge method presented in the lecture, on each of the following sets of ordered files:

a. m=n and a[i] < b[i] < a[i+1], e.g. a=[6, 9, 12, 15, 29, 37] and b=[8, 10, 14, 25, 33, 45]

It n+m-1 comparisons because the smaller value is sent to output.

b. m=n and a[n] < b[1], e.g. a=[2, 5, 9] and b=[12, 14, 16]

a[i] refers the value in position i of file a, etc.

Here the number of comparisons is n.

4. Determine the number of comparisons (as a function of n and m) that are performed in merging two ordered files a and b of sizes n and m, respectively, by the merge method presented in the lecture, on each of the following sets of ordered files:

a. m=n and a[n/2] < b[1] < b[m] < a[(n/2)+1], e.g. a=[2, 5, 7, 55, 61, 72] and b=[9, 15, 17, 21, 29, 46]

It is $\frac{n}{2} + m$ comparisons.

b. m=1 and b[1] < a[1]

Here one comparison occurs. But, here the question has an error, b shows the second index and m = 1.

c. m=1 and a[n] < b[1], a[i] refers the value in position i of file a, etc.

Here there are n comparisons. The same error exists in the question though...