Sorting The Rows of a Matrix Preserves the Sorted Columns

Jayadev Misra

2/19/96

Problem: Given is a matrix whose columns are sorted. Show that if each row is sorted individually then the columns remain sorted. Dijkstra expresses this condition by

$$(p \leq q) \Rightarrow [(sorted\ p) \leq (sorted\ q)]$$

Without loss in generality we will consider a matrix having 2 columns.

Proof 1: Consider a sorting algorithm, such as bubble sort, that compares and exchanges pairs of items. We apply the algorithm in parallel to both rows; at each step the algorithm chooses a pair of indices i, j and compares the items at i, j in each row. The result of comparison is a flip – exchange the items – or a no-op. We show that each step maintains the sortedness of the columns.
If items in both rows are flipped or neither is then the sortedness is maintained. Otherwise, let the top row have elements a, b and the bottom row have c, d in positions i, j. Suppose c, d are flipped whereas a, b are not; we show that $a \leq d$ and $b \leq c$.
{Since a, b are not flipped} $a \leq b$
{columns were sorted} $b \leq d$
{transitivity} $a \leq d$.
Similarly, {columns were sorted} $b \leq d$
{Since c, d are flipped} $d \leq c$
{transitivity} $b \leq c$.

Proof 2: Let x be the value of an item in the top row. We show that after sorting, the item below it is at least as large as x. Let the number of items in the top row having values at least x be n. Then x occupies one of the last n positions in its row after sorting. The number of items in the bottom row that are at least x is greater than or equal to n, because each item of the top row has a distinct non-smaller item in the bottom row. Therefore after sorting, the last n entries of the bottom row are at least x.