IOI’12 Ideal city Solution

Simple solutions use Floyd-Warshall algorithm or iterated BFS on the unary-cost edges, and both require O(N) space: time is O(N^3) for Floyd-Warshall, and O(N^2) for the iterated BFS, which requires N times the number O(N) of edges.

A more efficient solution is the following one.

- For every row r, consider the connected groups of cells on row r; each such group becomes a node of a tree, with a weight corresponding to the cardinality of the group. Two nodes of this tree are adjacent iff there are at least two cells in the corresponding groups sharing a common edge. Repeat the same argument for every column c.
- The above description yields two node-weighted trees, one (let us call it TH) corresponding to horizontal node-groups and another (TV) for vertical node-groups.
- Now, a shortest path between any two cells can be decomposed into two shortest paths along TV and TH: the two corresponding integers are called the vertical and horizontal contribution, respectively.
- Let us limit ourselves to the horizontal contributions. The sum of all horizontal contributions can be computed as the sum of w(x)*w(y)*d(x,y) over all possible distinct pairs of distinct nodes x and y in TV: here, w(x) and w(y) are their weight (number of cells) and d(x,y) is their distance in TV.
- The latter summation can be computed in linear time in the number of edges of TV, by observing that it is equivalent to the sum of S(e)*S’(e) over all edges e of TV, where S(e) and S’(e) are the sum of the weights of the two components of the tree obtained after removing the edge e.