Looting Byteland - Editorial

Difficulty:

Hard

Prerequisites:

Graph - (Tutorial)
Max Flow - (Tutorial)
Min Cost Flow - (Tutorial)

Problem in Brief:

Given an N * N grid A with each cell containing a value . Find K paths from (1, 1) to (N, N) such that the sum of cells covered by at least 1 of the paths is maximized.

Editorial:

This problem can be solved using Minimum cost Max flow. It's intuitive to consider that greedy solution works but we'll use an example to show why It's wrong.
Firstly, to solve this problem using MCMF we will create a new graph using the given grid. The new graph will contain 2 * N * N nodes. We will add edges in the graph using following scheme.
 Each edge is of the form (u, v, cap, cost) -> means there is an edge from u to v with capacity = capacity = cap and cost = cost. Let's add the following edges for all nodes (i, j) corresponding to jth cell of the ith row in the grid:

((i, j), (i, j + 1), INF, 0)
Where INF is a really large number
((i, j), (i + 1, j), INF, 0)

Now for each cell (i, j) make an IN vertex and an OUT vertex and add the following edges:

(IN(i, j), OUT(i, j), 1, -A[i][j])

(IN(i, j), OUT(i, j), K - 1, 0)

Note: Be careful to not go out of boundary and correctly connect the Out type of vertices to In type of vertices in first 2 type of edges

Now run the MCMF algorithm on this graph and multiply it with -1 to get the answer. It is easy to see that A[i][j] will only be added once and will only be added if it’s in a path.

Let’s work through an example to see why greedy is wrong:

Consider the testcase:

```
4 2

5 5 1 1
1 0 0 1
1 0 0 1
1 1 5 5
```

The Greedy algorithm(select best path at each iteration for K iterations) returns 25 whereas the answer is 28

If we build the graph using the above described algorithm then it will return the cost -28 which multiplied by -1 gives us the correct answer.

Looting Byteland - Editorial

Time Complexity:

The Min cost Max flow algorithm is the most expensive operation and it's Time Complexity is
O(N^4 * Maxflow)

Similar Problems:

First
Second
Third