###

**36th Annual High School Programming Contest**

| **Sponsored by** |  |
| --- | --- |

##### April 12, 2024

###### Gold Problem #7:  The Transfinder Problem

Background Information: Transportation systems generate a variety of challenging problems. School bus routes, police patrols, mail delivery, and trash collection are examples of systems that generally look for complete coverage while avoiding overlap. Many of these applications are connected to famous puzzles like the Konigsberg Bridge problem, the traveling salesperson problem, and the number link puzzle. The number link puzzle is a routing challenge that requires finding non-intersecting paths that connect matching numbers on an M x N rectangular grid. The object is to connect the matching numbers with “lines” connecting matching numbers at the endpoints. In addition, all spaces on the grid must be covered exactly once. Lines are vertical and horizontal segments.

To develop software to solve the puzzle it is common to index the grid so that squares can be referenced. Row/column indices start at 0, with (0, 0) representing the upper left corner. A puzzle and its solution with indices are shown on the right. 

Note that in this example, there are no endpoints that are adjacent to each other. Endpoints that are adjacent to each other may or may not be connected to each other with horizontal and vertical segments.

###### Programming Problem:

Input:  On the first line, the number of rows (R) and columns (C) (with R and C ≤ 7) of the grid followed by the number K (≤ half the number of squares in the grid) of numbers to match. The first line of input will be followed by K lines, each line containing starting row/column coordinates for a starting number and an ending number.

Output: The R rows with C symbols in each row. The symbols consist of the numerical input and lowercase letters indicating parts of the line segments used to connect endpoints. The path between endpoints 1 is indicated with “a”, between endpoints 2 with “b”, and so on. If there is no solution, then just print NO SOLUTION (in uppercase).

###### Example I/O for the puzzle and solution above:

######  Input: Output:

7 7 5 bbb4ddd

5 2 2 4 b3bb25d

6 0 1 4 bcc31ed

1 1 2 3 bee5aed

0 3 6 4 beaaaed

1 5 3 3 be1eeed

2eee4dd