Q 2.1)

**Subproblems:** for each and , Let be the problem of determining , the maximum number of possible unique paths to reach from with only two kinds of moves (down one cell or right one cell).

**Recurrence:** for and ,

At any point that interpreting no possible paths through that cell.

There are only two possible ways to reach any cell which is either from its top cell or its left cell since only moving down one cell or right one cell are allowed. So, the number of possible unique paths to reach any cell is the sum of possible unique paths of its top cell and its left cell.

Since depends on (its top cell) and (its left cell), we solve subproblems in increasing order of then .

**Base cases:**

If or then

Interpreting no possible paths to outside the whole cells.

Interpreting only one possible path to the start which is staying at the start unless there is a box.

If , it means the warehouse layout meets the requirement for having a way to reach the exit with only two kinds of moves. The warehouse layout doesn’t meet the requirement otherwise.

Overall time complexity is for iterating through all elements in a 2D array of size (all elements from one column then move to another column). And each of subproblems is solved in constant time.

Q 2.2)

**Subproblems:** for each and , Let be the problem of determining , the minimum number of boxes that must be removed in order to reach from with only two kinds of moves (down one cell or right one cell).

**Recurrence:** for and ,

At any point that interpreting no boxes need to be removed to make a path to that cell.

There are only two possible ways to reach any cell which is either from its top cell or its left cell since only moving down one cell or right one cell are allowed.

The number of boxes that must be removed in order to reach any cell is either from its top or left cell then increase it by 1 if the cell has a box on it (since that box must be removed in order to reach the cell).

So, the smallest number could be found by choosing the minimum number of boxes that must be removed in order to reach its top and its left cell (among two of them).

Since depends on (its top cell) and (its left cell), we solve subproblems in increasing order of then .

**Base cases:**

If or then

Interpreting no possible paths to outside the whole cells no matters how many boxes are removed.

.

Interpreting a box must be removed if it happens to be at the starting cell.

is the smallest number of boxes that must be removed to meet the requirement.

Overall time complexity is for iterating through all elements in a 2D array of size (all elements from one column then move to another column).

And each of subproblems is solved in constant time.

Q 2.3)

Since no matter what unique paths we take, we would need to move down one cell ( to ) then move to the right one cell ( to ) at least once (not in a strict order) creating a corner ( to ) in a path, in order to reach the exit at the bottom-right from top-left (since ).

A shortcut is when we move down one cell and to the right one cell at the same time ( to ). That means whenever we have a corner, we would be able to use a shortcut instead because we would end up in the same destination location (and as mentioned above that there’s always at least one corner in a path).

So, if there exists a path to the exit, there would have at least one corner that could be using a shortcut.

Q 2.4)

**Subproblems:** for each , and , let be the problem of determining , the minimum sum of hazard rating of a path from to taking exactly shortcuts with only two kinds of moves, and , the minimum sum of hazard rating of the path taking only one shortcut from to .

**Recurrence:** for , and ,

There are only two possible ways to reach any cell which is either from its top cell or its left cell since only moving down one cell or right one cell are allowed. So, taking the path from the cell that has less minimum sum of hazard rating would make it become the minimum one.

Using of its top-left cell adding this cell’s hazard rating to this cell is the minimum sum of hazard rating taking shortcut to this cell. However, in order to assure that only one shortcut could be taken, if then it means that a necessary shortcut had been taken to pass through boxes and no further shortcuts from that cell could be taken anymore. Or that the path is blocked by a box.

There are three ways to reach a cell if a shortcut could be taken, via its top-left cell, top cell and left cell. Then the minimum among them is the minimum sum of hazard rating of a path from to taking exactly one shortcut. This is choosing between the path that takes a recent shortcut, or the paths that took a shortcut on its way.

Since depends on (its top cell) and (its left cell). Same goes for but it also depends on which depends on . So, we solve subproblems in increasing order of then then .

**Base cases:** if or then . .

Interpreting no possible paths from outside the grid and the hazard rating at the starting cell is the only option.

If of then and .

Interpreting shortcuts couldn’t be taken since there’s no corners appear from only moving right one cell or only moving left one cell.

Whenever there’s a box in a cell, no paths are possible to reach that cell, so and are .

is the minimum sum of hazard rating of a path from start to exit taking exactly one shortcut with only two kinds of moves. There would always be at least one valid to make this solution possible since the warehouse always has a valid path that could take a shortcut as explained in Q2.3.

Overall time complexity is for iterating through each element in a 2D array of size twice (all elements from one column then move to another column). And each of subproblems is solved in constant time. Costing in total.