



Daily Coding Problem

Good morning! Here's your **coding** interview **problem** for today.

This **question** was asked by Zillow.

You are given a 2-d `matrix` where each cell represents number of coins in that cell. Assuming we start at `matrix[0][0]`, and can only move right or down, find the maximum number of coins you can collect by the bottom right corner.

For example, in this matrix

```
0 3 1 1
2 0 0 4
1 5 3 1
```

The most we can collect is $0 + 2 + 1 + 5 + 3 + 1 = 12$ coins.

Create a grid as above a 2D integer array.....

Can try to look at different way...

generate numbers between 1 to 3 (3 is maximum move across `grid[0].length` and 2 is the maximum move across `grid.length`)...

we know maximum moves that can be taken is 5 (as shown above).....

Since it can move down and right, it can complete a staircase motion... 1 step at a time

This would be (1,1,1,1,1)

Get all the combinations of total = 5... This would be **combination with replacement**.....

We also know the singular movements are 1, 2, 3 (either down or right).

These are only possibilities..

So required to do $C(3,5)$ - this can not be negotiated since it is based off the combinations generated via Combinations class).

Fill the HashSet (which will keep unique combinations) with these combinations.... This will be part of a do while loop where `set.size < combinations`...

Each set entry can be such as 1,1,1,1,1 1,2,2 (can be stored in a string or integer array).

Now go through each entry....

Need to remember that there is alternation between moving right and down....

This is a bit difficult to understand but I am attempting to explain it...

If there is a single right movement and another right movement, this would be covered if the first move was [2, X, X, X]. Alternatively if there were three single 1 x right movement (this would be covered by [3, X, X, X, X].... The larger move will always supersede the smaller move....

This logic holds since moves are in two directions.. It would fail in other circumstances..

So each entry in the set has to be tested both ways...

For instance 1=right,1=down,1=right,1=down,1=right or

1=down,1=right,1=down,1=right,1=down

Transpose these movements onto the 2D grid array above...

Might need try and catch since it could go beyond the boundaries....

If there are `ArrayIndexOutOfBoundsException`, discard this entry from the set and move on.

For any successful finishes, can use `stingjoiner` and `stringbuilder` to concatenate and create `String` for (set entry and also the coins collected)

Traverse through the repository (can be `String` array), extract the coins collected, find the maximum coins collected and output the entry to end user...

Again, there might be multiple paths with same highest coins.

So, it is required to store indexes of the array in which it has hit maximum coins collected.

Present these to end user....