Note that for a given `(r, c)`

the following code will not be executed more than once :

```
memo[r][c] =  V[r][c] + min(findMinPath(V, r + 1, c), findMinPath(V, r, c + 1));
```

Once memo[r][c] is set, the functions will return at

```
if (memo[r][c] != -1) return memo[r][c];
```

So, every function ends up calling other functions at most 1 time.
In other words, every function ends up executing atmost O(1) times (Note that you can shift the part about checking for memo[r][c] != -1 at the callsite).

`O(R * C)` possible number of combinations are possible for `(r, c)`
Hence, the time complexity of the function : O(R*C)