

What is the worst case time complexity of the following code:

```
int memo[101][101];
int findMinPath(vector<vector<int> >& V, int r, int c) {
    int R = V.size();
    int C = V[0].size();
    if (r >= R || c >= C) return 100000000; // Infinity
    if (r == R - 1 && c == C - 1) return 0;
    if (memo[r][c] != -1) return memo[r][c];
    memo[r][c] = V[r][c] + min(findMinPath(V, r + 1, c), findMinPath(V, r, c + 1));
    return memo[r][c];
}
```

Callsite :

```
memset(memo, -1, sizeof(memo));
findMinPath(V, 0, 0);
```

Assume  $R = V.size()$  and  $C = V[0].size()$  and  $V$  has positive elements