Starting at top-left of nxm grid, moving only down or right, how many ways to reach bottom-right?

Starting at top-left of nxm grid, moving only down or right, how many ways to reach bottom-right?

\[ A[r,c] \]

repetitive subproblems
want to avoid repetition

\[ A[r-1,c] \]

\[ A[r,c-1] \]

\[ [r-1,c-1] \]

\[ [r-2,c-1] \]

\[ [r-2,c] \]

\[ [r-3,c] \]

\[ [r,c] \]

\[ [r,c-1] \]

\[ [r,c-2] \]

\[ [r-1,c-2] \]

\[ [r-2,c-2] \]

\[ [r-1,c-2] \]

\[ [r-1,c-3] \]

\[ [r,c-3] \]

min\{r,c\} full levels
\[ \Omega(2^n) \] for nxn

[1,c]
Starting at top-left of $n \times m$ grid, moving only down or right, how many ways to reach bottom-right?
How many times will we recurse in a unique way?

\[ A[r,c] \]

\[ \rightarrow r \cdot c \text{ distinct subproblems} \]

- \[ A[r-1,c] \]
  - \[ A[r-2,c] \]
    - \[ A[r-3,c] \]
      - \[ A[r-3,c-1] \]
      - \[ A[r-3,c-2] \]
  - \[ A[r-2,c-1] \]
  - \[ A[r-2,c-2] \]
  - \[ A[r-2,c-3] \]
- \[ A[r-1,c-1] \]
  - \[ A[r-1,c-2] \]
  - \[ A[r-1,c-3] \]
- \[ A[r,c-1] \]
  - \[ A[r,c-2] \]
  - \[ A[r,c-3] \]

how many times will we realize that we have seen a subproblem before?
MEMOIZATION (making memos)

For this problem, \( m \times n \) table

\[
\]

Recursion:
- First find \( A[r-1,c] \) up
- Then find \( A[r,c-1] \) left

\( A[r,c] \)
**MEMOIZATION**  (making memos)

For this problem, m x n table


<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Recursion:**

- first find \( A[r-1,c] \) \( \uparrow \)
- then find \( A[r,c-1] \) \( \leftarrow \)
MEMOIZATION (making memos)

For this problem, \( m \times n \) table

\[
\]

Recursion:
- First find \( A[r-1, c] \)
- Then find \( A[r, c-1] \)

\( \Theta(n \cdot m) \) time & space
Starting at top-left of nxm grid, moving only down or right, how many ways to reach bottom-right?

**DYNAMIC PROGRAMMING** (bottom-up: base cases first)


Fill any cell as long as what it depends on is full.
Starting at top-left of nxm grid, moving only down or right, how many ways to reach bottom-right?

**Dynamic Programming** (bottom-up: base cases first)

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A\[r,c\] = A\[r-1,c\] + A\[r,c-1\]

*Fill any cell as long as what it depends on is full*
Starting at top-left of \( nxm \) grid, moving only down or right, how many ways to reach bottom-right?

**Dynamic Programming** (bottom-up: base cases first)

\[
\begin{array}{cccccccc}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
1 & 3 & 6 & & & & & \\
1 & 4 & 10 & & & & & \\
1 & 5 & & & & & & \\
1 & 6 & & & & & & \\
\end{array}
\]

Starting at top-left of nxm grid, moving only down or right, how many ways to reach bottom-right?

**DYNAMIC PROGRAMMING** (bottom-up: base cases first)

\[
\]

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
<td>6</td>
<td>10</td>
<td>15</td>
<td>21</td>
<td>28</td>
<td>36</td>
<td>45</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>4</td>
<td>10</td>
<td>20</td>
<td>35</td>
<td>56</td>
<td>84</td>
<td>120</td>
<td>165</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>5</td>
<td>15</td>
<td>35</td>
<td>70</td>
<td>126</td>
<td>210</td>
<td>330</td>
<td>495</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>6</td>
<td>21</td>
<td>56</td>
<td>126</td>
<td>252</td>
<td>362</td>
<td>692</td>
<td>1187</td>
</tr>
</tbody>
</table>

fill any cell as long as what it depends on is full.
Starting at top-left of nxm grid, moving only down or right, how many ways to reach bottom-right? ... with obstacles

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>O</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>O</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>O</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>O</td>
<td>O</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>O</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>O</td>
<td></td>
</tr>
</tbody>
</table>

There are 4 obstacles in the grid.
Starting at top-left of nxm grid, moving only down or right, how many ways to reach bottom-right? ... with obstacles
Starting at top-left of nxm grid, moving only down or right, how many ways to reach bottom-right? ... with obstacles

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>6</td>
<td>14</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>5</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>20</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>6</td>
<td>6</td>
<td>7</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>15</td>
<td>35</td>
</tr>
</tbody>
</table>