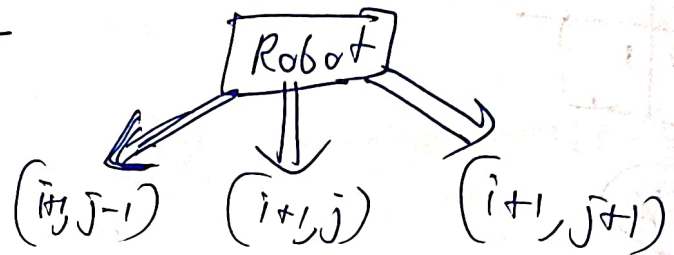


Any Robot has 3 choices.



So intuitively my brain thought, Starting from first row, to last row.

Robot 1 has 3 options and Robot 2 has 3 options,

3 & 3 → 9 options, → given. $(R_{1x}, R_{1y}, R_{2x}, R_{2y})$

(R_{1x}, R_{1y}) → Robot 1's position

(R_{2x}, R_{2y}) → Robot 2's position

So I store state as. $\{ [R_{1x}] [R_{1y}] [R_{2x}] [R_{2y}] \}$

Number of cols ≤ 70 .

So total complexity was.

$$(70 \times 70 \times 70 \times 70) \times \underline{\underline{9}}$$

↓
Combinations
we are making

this was making TLE,

then I realized, one more step can be further reduced

↓
(state)

Since both robots are moving downwards at the same row together, so we don't need to keep two rows of both the robots.

cache [row] [col1] [col2]
 ↓ ↓ ↓
 robot1 robot2
 both robots

New Complexity is

$$(70 \times 70 \times 70) \times 9$$

This worked.