Here we if we use backtracking, TLE
happens as the input size is big
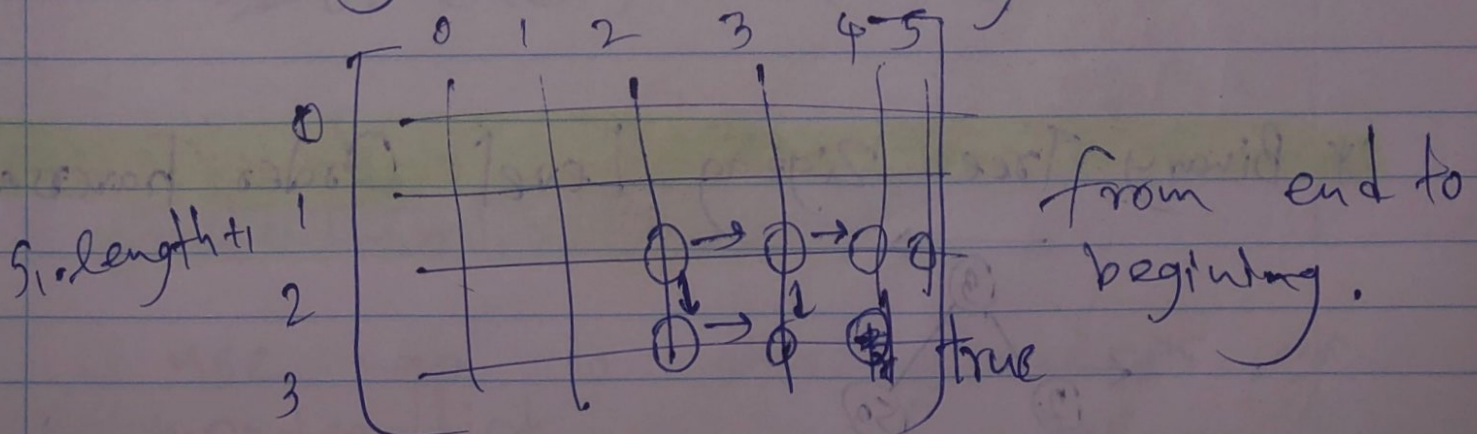
$$O\left(2^{m+n}\right)$$

$O(m+n)$ for recursive call stack

Recursive with memorization, can be
used

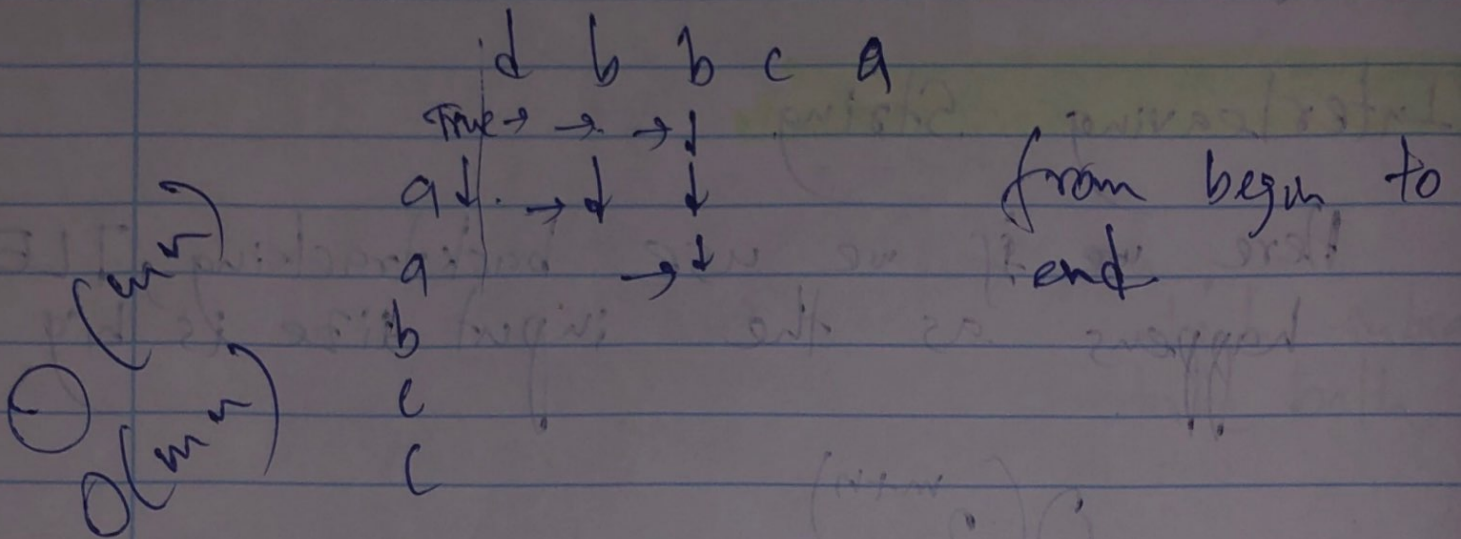To store posibilities use Boolean[][]

$$O(m \cdot n)$$
$$O(m \cdot n)$$

Using Dynamic Programming 2D

array

$s_2.length + 1$

	0	1	2	3	4 5
0					
1					
2					
3					

$s_1.length + 1$

from end to
begining.

true

- using 2D DP solution

$$d \quad b \quad b \quad c \quad a$$

True → → → ↓
a ↓ → ↓ ↓
a → ↓
b
c
c

from begin to end

a a d b b c b c a c

as an optimization, we can use 1D Array & change in place as current value represents previous value

$$\bigcirc (m \cdot n)$$

$$\bigcirc (n)$$

$$\bigcirc (m \cdot n)$$

$$\bigcirc (m \cdot n)$$