Problem 2 – Paths

You are given a matrix of directions. **The rows are zero-based.** The possible directions are four:

- "dr" stands for "down-right" direction
- "ur" stands for "up-right" direction
- "ul" stands for "up-left" direction
- "dl" stands for "down-left" direction

You should generate another matrix with the same size. The numbers in each row are consecutive. The leftmost (first) number in each row is a power of 2, calculated with the formula $\mathbf{2^{row}}$ where **row** is the number of this row.

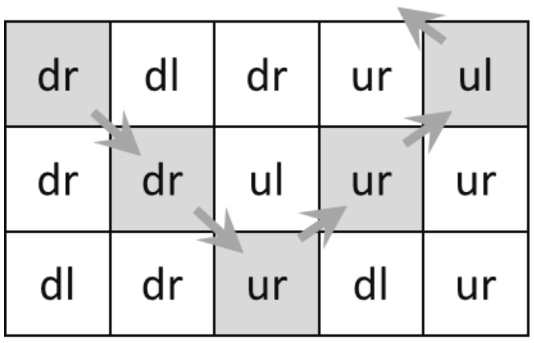
Your task is to find the sum that can be found, using the given directions and summing the numbers in each cell you step into. If with the current direction, you step out of the matrix – print "**successed with SUM**", where sum is the calculated sum. If you step on a cell, that you have **previously stepped in** – print "**failed at POSITION**", where POSITION is the position of the previously visited cell.

You start always at position (0, 0) – row = 0, column = 0.

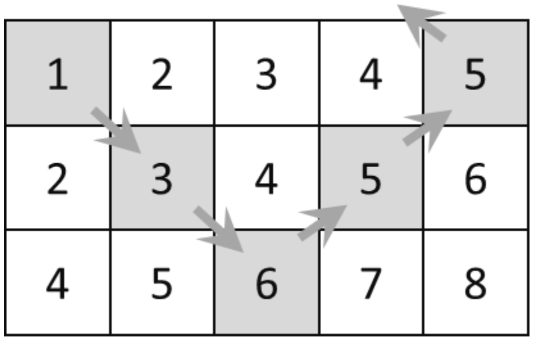
Create a function in JavaScript that solves this problem.

Example:

The **sum is 20,** the output should be: **"successed with 20"**

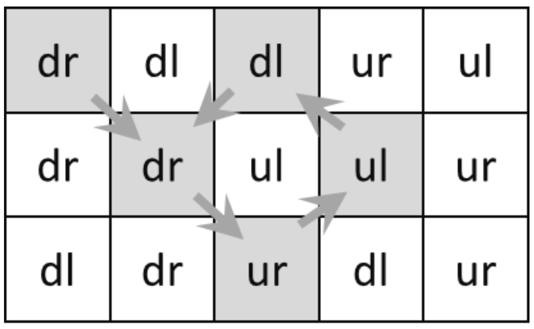


dr	dl	dr	ur	ul
dr	dr	ul	ur	ur
dl	dr	ur	dl	ur



1	2	3	4	5
2	3	4	5	6
4	5	6	7	8

Stepped on a **visited** cell **with coordinates (1, 1**), the output should be: **"failed at (1, 1)"**



dr	dl	dl	ur	ul
dr	dr	ul	ul	ur
dl	dr	ur	dl	ur

Input

The first value of the input arguments you will be the numbers R and C, separated by a space

- R is the number of rows in the matrix
- C is the number of columns in the matrix

The next R values in the input arguments will be exactly C directions, separated by a space

Output

Your function should return a single string:

- If **success**, return **"successed with SUM"**, where sum is the sum of the visited cells
- If **fail**, return **"failed at POSITION"**, where POSITION is the position of the already visited cell

Constraints

- **N will be between 2 and 30, inclusive**
- **C will be between 2 and 1000, inclusive**
- Allowed working time for your program: 0.1 seconds.
- Allowed memory: 16 MB.

Examples

Input	Output	Explanation

`args =[`
`  '3 5',`
`  'dr dl dr ur ul',`
`  'dr dr ul ur ur',`
`  'dl dr ur dl ur'`
`]`
	`successed with 20`	`Explained in the example above`

Input	Output	Explanation

`args = [`
`  '3 5',`
`  'dr dl dl ur ul',`
`  'dr dr ul ul ur',`
`  'dl dr ur dl ur'`
`]`
	`failed at (1, 1)`	`Explained in the example above`