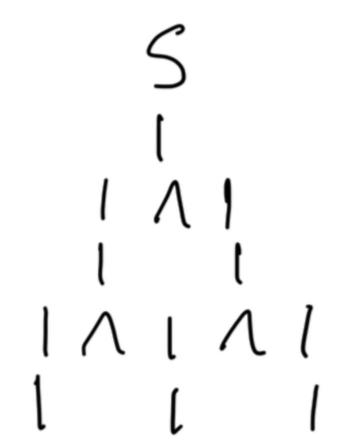
Part 1

ex:  → $\boxed{3}$ splits, i.e. # of $\boxed{\begin{matrix} | \\ \wedge \end{matrix}}$.

alg: $f(A) = f(\text{beams}^0, A[1:])$

$= \text{nsplits}(\text{beams}^0, A[1]) + f(\text{beams}^1, A[2:])$

iter i:

$\text{beams}^i = [\dots, 0, 1, 0 \dots]$ where 1 → beam

$A[i+1] = [\dots, 0, 1, 1, \dots]$ where 1 → splitter

$\text{beams}^{i+1} = [\dots, 1, 0, 1, \dots]$,

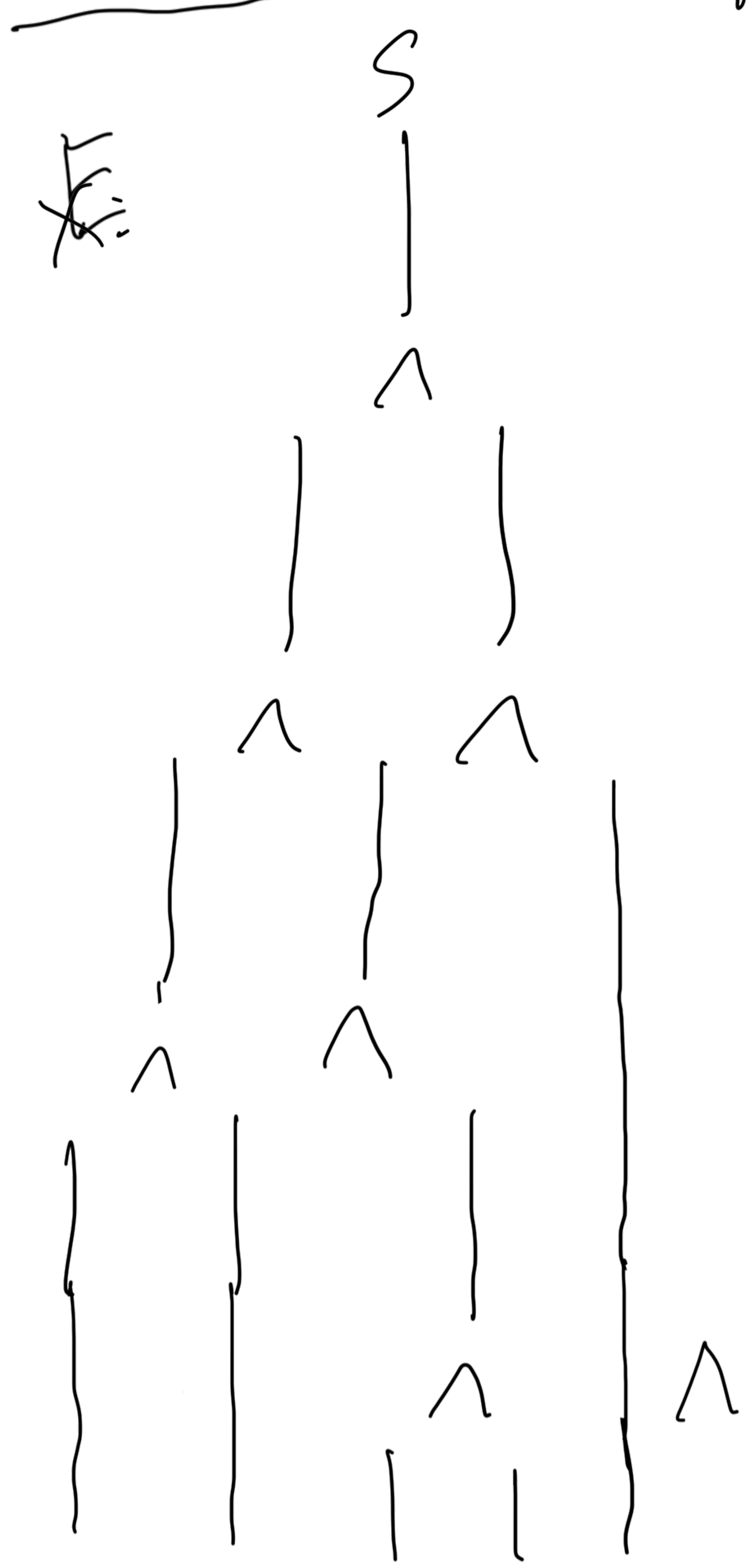
i.e. beams^i w/ splits applied.

if A is $m \times n$, above alg is $O(m \times n)$

Since you look at each row once, doing one scan.

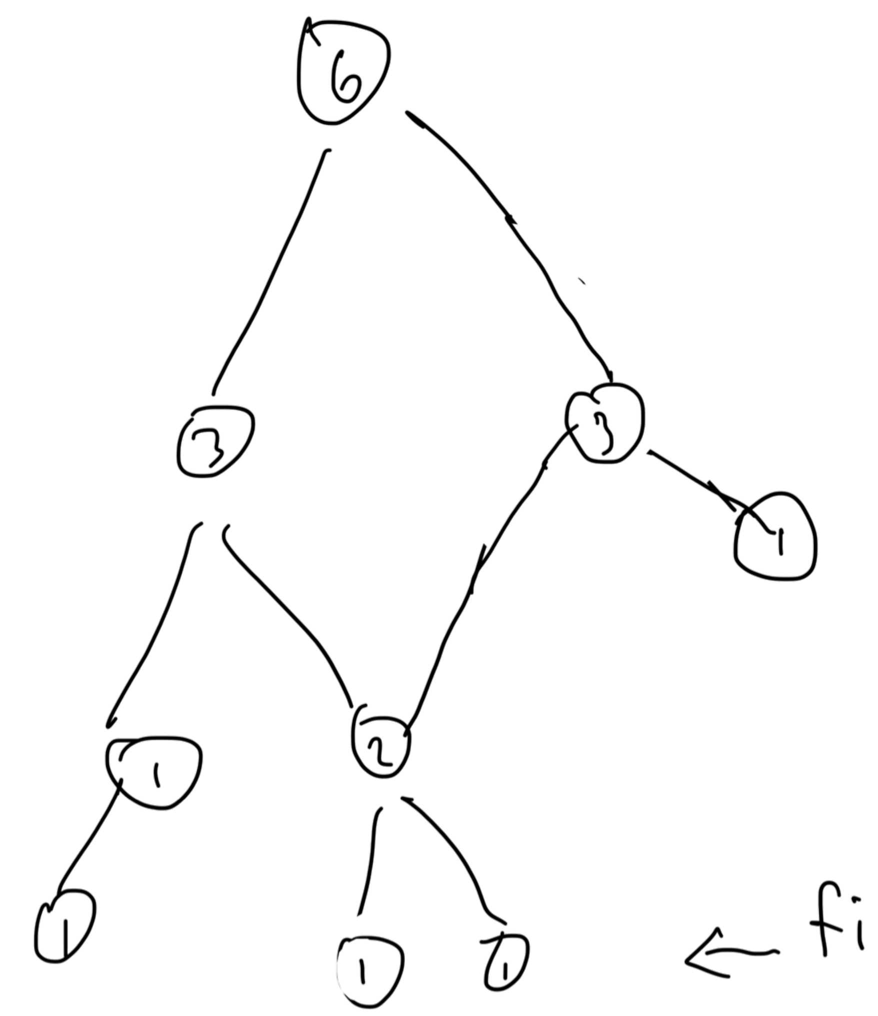
Part 2

~~Ex:~~



$f(A)$ = # of unique possible paths through A.

Can we model as a tree?

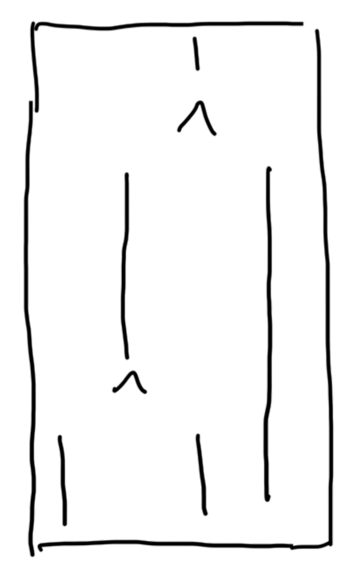


← fill in from bottom-top

if so, it's easy: DFS.

or BFS - keep track of all paths at each level of tree.

Idea: do pictured bottom-up propagation, but on a grid. e.g.

 ⟷