Edit distance between two strings: min # of insertions, deletions, + substitutions to turn one string into the other.

FOOD → MOOD → MOND → MONED → MONEY

so ED ≤ 4.

F	O	O		D
↓	↓	↓	↓	↓
M	O	N	E	Y
sub	–	sub	ins	sub

can overlap strings to easily picture each edit

Given strings $A[1..m]$ &

$B[1..n]$.

Goal: Compute edit distance between A & B

1) Find the recursive strategy:

If we guess the rightmost edit operation...

recurse on what's (to the) left!

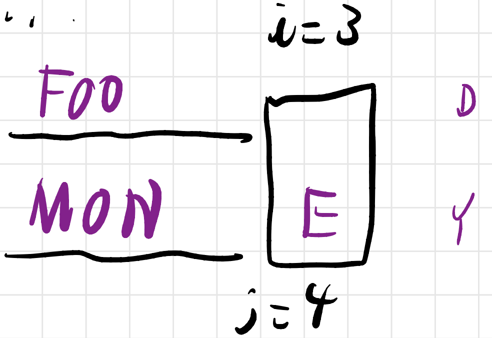
$\text{Edit}(i,j)$: edit distance between $A[1..i]$ & $B[1..j]$.

Ultimately, we want to compute...

~~A~~ $\text{Edit}(m,n)$.

$\text{Edit}(i,j) = \ldots$

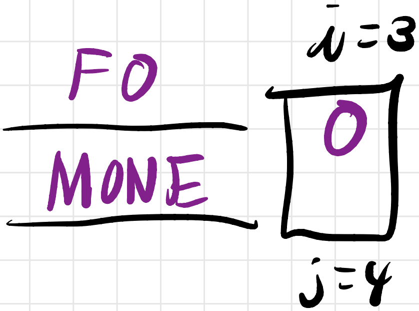
Insertion:



$$\underset{\text{for the insertion}}{1} + \text{Edit}(i, j-1)$$

want to change $A[1..i]$ into $B[1..j-1]$ then insert $B[j]$

Deletion



$1+\text{Edit}(i-1, j)$

change $A[1..i-1]$ into $B[1..j]$ + delete $A[i]$.

Substitution (?)

if last characters differ

$1+\text{Edit}(i-1, j-1)$

o.w.

$\text{Edit}(i-1, j-1)$

[illegible]

$$Edit(0,j)=j$$

$$Edit(i,0)=i$$

Double check: $Edit(0,0)=0$

Given proposition P,

let $[P]=1$ if P is true

0 o.w.

$$Edit(i,j)=\begin{cases} i & \text{if } j=0 \\ j & \text{if } i=0 \\ \min\left\{\begin{array}{c} Edit(i,j-1)+1 \\ Edit(i-1,j)+1 \\ Edit(i-1,j-1)+[A[i]\neq B[j]] \end{array}\right\} & \text{otherwise} \end{cases}$$

2) Dynamic Programming

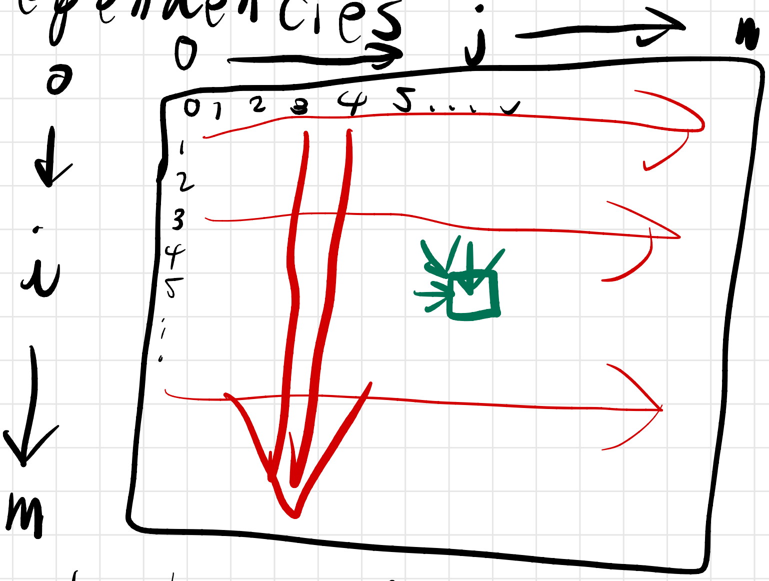
a) ~~sup~~ subproblems:

$\underline{0} \le i \le \underline{m}$ $\qquad$ $\underline{0} \le j \le \underline{n}$

b) memoization: store answers in a 2D array

Edit[0..m, 0..n]

c) dependencies

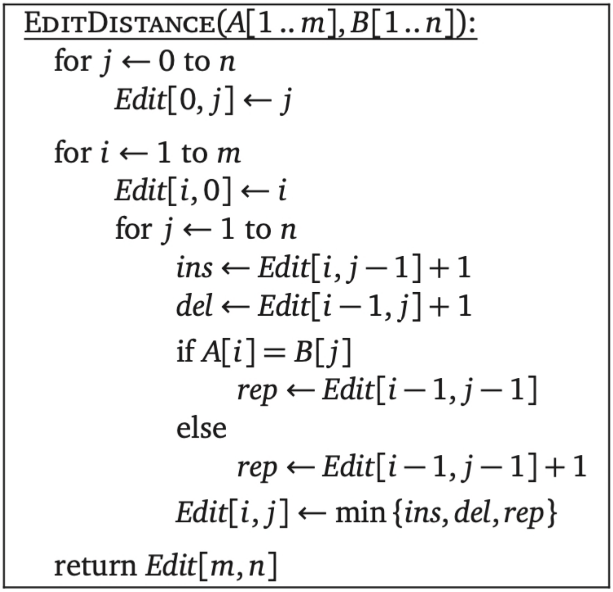


d) evaluation order

e) space: $\underline{O(mn)}$

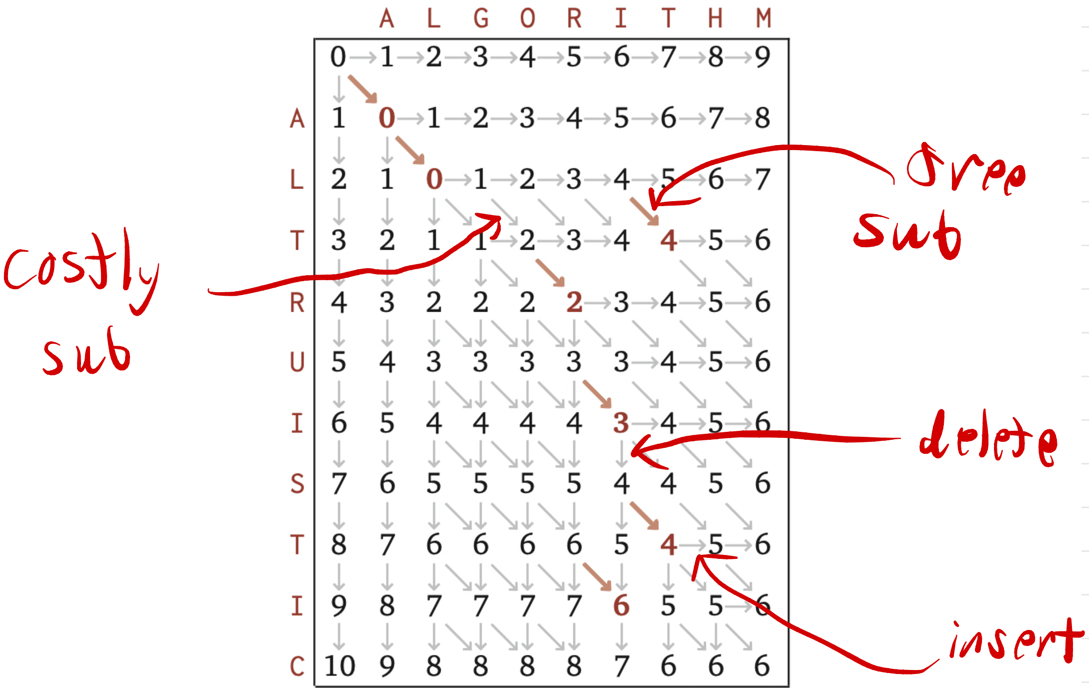
time: $O(1) \cdot O(mn) = \underline{O(mn)}$

f) code!



```
EditDistance($A[1..m], B[1..n]$):
    for $j \leftarrow 0$ to $n$
        $Edit[0, j] \leftarrow j$
    for $i \leftarrow 1$ to $m$
        $Edit[i, 0] \leftarrow i$
        for $j \leftarrow 1$ to $n$
            $ins \leftarrow Edit[i, j-1] + 1$
            $del \leftarrow Edit[i-1, j] + 1$
            if $A[i] = B[j]$
                $rep \leftarrow Edit[i-1, j-1]$
            else
                $rep \leftarrow Edit[i-1, j-1] + 1$
            $Edit[i, j] \leftarrow \min\{ins, del, rep\}$
    return $Edit[m, n]$
```

[Wagner, Fischer '74]



		A	L	G	O	R	I	T	H	M
	0	1	2	3	4	5	6	7	8	9
A	1	0	1	2	3	4	5	6	7	8
L	2	1	0	1	2	3	4	5	6	7
T	3	2	1	1	2	3	4	4	5	6
R	4	3	2	2	2	2	3	4	5	6
U	5	4	3	3	3	3	3	4	5	6
I	6	5	4	4	4	4	3	4	5	6
S	7	6	5	5	5	5	4	4	5	6
T	8	7	6	6	6	6	5	4	5	6
I	9	8	7	7	7	7	6	5	5	6
C	10	9	8	8	8	8	7	6	6	6

$O(m+n)$ additional time to output optimal sequence of edits

No $O(n^{1.999})$ time algorithm (assuming a certain complexity conjecture)