

HW 6: Today Tue Oct 13

GPS 7: Mon Oct 19

Midterm 1 grades!

Extra office hours: Wed 11am to 12pm
Thur 2pm to 3pm

2336 Siebel

Drop deadline Fri Oct 16.

Edit distance:

Given two arrays of characters $A[1..m]$ &
 $B[1..n]$.

The edit distance between A & B is

the minimum number of

insertions,
deletions, or
substitutions

of single characters to transform A
into B .

A L G O R I T H M $\rightarrow$ A L T R U I S T I C

A L O R ^u I H M

edit distance

A L O R V I ^s T H M

< 6

A L O R V I S T H M

A L O R V I S T I M

A L O R V I S T I C

A L T R V I S T I C

Plan?: Guess an edit & recurse.

- may increase length of string!

- could lead to many many subproblems

A L G O R I T H M

A L T R V I S T I C



deletion



insertion



substitution

↑ two row representation of all the edits.
Try to decide what each column should
be?

Try to decide each column right to

left.

ALGOR	{	I	T	H	M	previous decisions
ALTRV		I	S	T	I	

next columns?

want their edit distance!

+ they are prefixes of input

can describe prefixes with one index
each

$Edit(i, j)$: Edit distance between

$A[1..i] + B[1..j]$. ← closed

Form definition!

Want to compute $Edit(m, n)$.

Want to compute $Edit(i, j)$.

What is the last column/edit to
turn $A[1..i]$ into $B[1..j]$?

Insertion?

ALGOR

ALTR

U

I T H M

I S T I C

$$Edit(i, j) = 1 + Edit(i, j-1)$$

Deletion?

ALGO

R

I

T

H

M

ALTRU

I S

T

I

C

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

Substitution or No edit:

ALGO

R

I

T

H

M

ALTR

U

I S

T

I

C

← costs 1

- or -

ALGOR

I

T

H

M

ALTRU I S T I C $\leftarrow \text{free}$

$[P] := 1$ if proposition P is true
0 if P is false

$$\text{Edit}(i, j) = [A[i] \neq B[j]] + \text{Edit}(i-1, j-1)$$

$$\text{Edit}(0, j) = j \quad (\text{have to do } j \text{ insertions})$$

$$\text{Edit}(i, 0) = i \quad (\text{have to do } i \text{ deletions})$$

(either imply $\text{Edit}(0, 0) = 0$)

$$\begin{aligned}
 \text{Edit}(i, j) = & \begin{cases} i & \text{if } j=0 \\ j & \text{if } i=0 \\ \min \begin{cases} \text{Edit}(i, j-1) + 1 \\ \text{Edit}(i-1, j) + 1 \\ \text{Edit}(i-1, j-1) + [A[i] \neq B[j]] \end{cases} & \text{o.w.} \end{cases}
 \end{aligned}$$

Subproblems: $0 \leq i \leq m, 0 \leq j \leq n$

Running Time: $O(1) \cdot O(mn) = O(mn)$

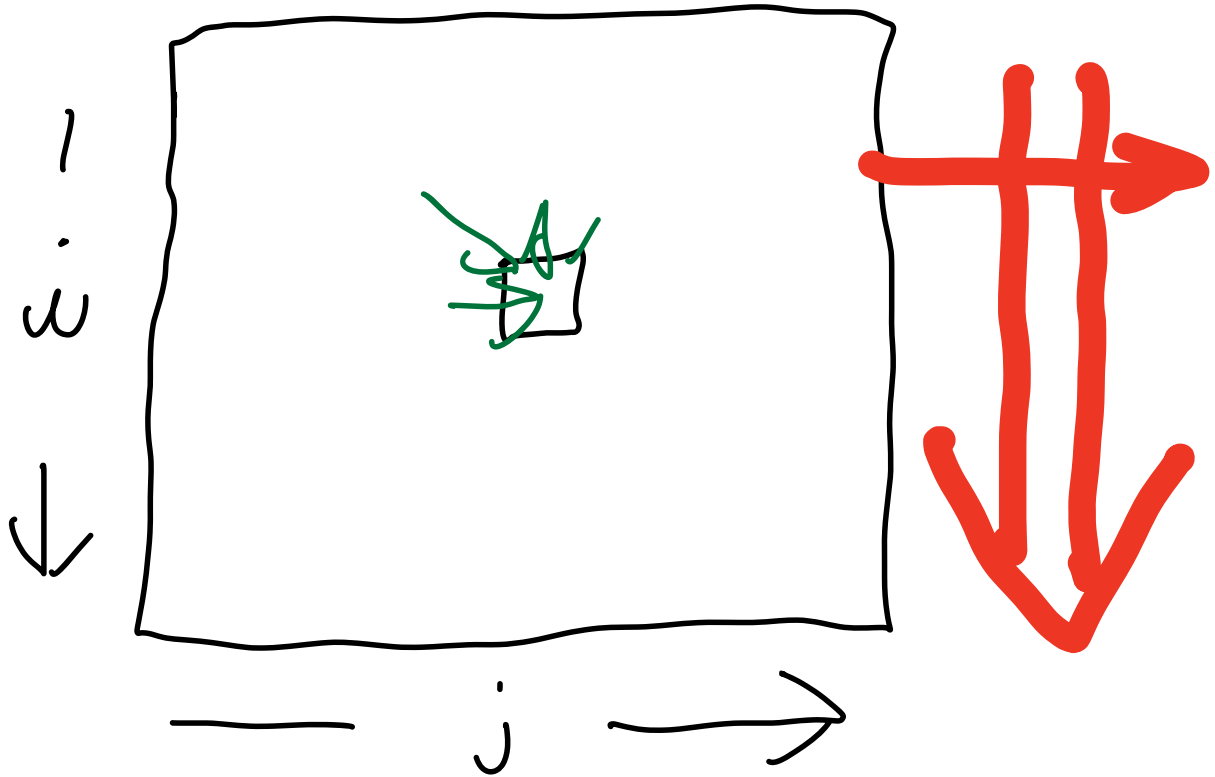
per / # subproblems

Memorization data structure: 2D array

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

Dependencies:

Eval order:



Done!!!

ok, have some code...

```
EditDistance( $A[1 \dots m], B[1 \dots n]$ ):
```

```
  for  $i \leftarrow 0$  to  $m$ 
```

```
    for  $j \leftarrow 0$  to  $n$ 
```

```
      if  $j = 0$ 
```

```
         $Edit[i, j] \leftarrow i$ 
```

```
      else if  $i = 0$ 
```

```
         $Edit[i, j] \leftarrow j$ 
```

```
      else
```

```
         $ins \leftarrow Edit[i, j - 1] + 1$ 
```

```
         $del \leftarrow Edit[i - 1, j] + 1$ 
```

```
        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  $Edit[m, n]$ 
```

		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

Trace best subproblem choices backwards from (m,n) to get best sequence of edits.

$O\left(\frac{mn}{\log(mn)}\right)$ "Method of Four Russians"

No $O(mn^{1-\epsilon})$ time algorithm for any

constant $\epsilon > 0$ assuming the

Strong Exponential Time Hypothesis
(SETH)