

# Edit Distance

Kwankamol Nongpong, Ph.D.

CSX3009 Algorithm Design

Department of Computer Science

Assumption University

# Minimum Edit Distance

- Given two strings  $A$  and  $B$  (i.e., two sequences of characters).
- Goal: Find the minimum number of edits to transform  $s_1$  to  $s_2$ .
- Edit operations include:
  - Insertion
  - Deletion
  - Substitution

# Example

F	O	O	D	
M	O	N	E	Y
S		S	S	I

4 edits

	F	O	O	D
M	O	N	E	Y
I	S	S	S	S

5 edits

F	O	O	D		
	M	O	N	E	Y
D	S		S	I	I

5 edits

F		O	O	D
M	O	N	E	Y
S	I	S	S	S

4 edits

F	O		O	D
M	O	N	E	Y
S		I	S	S

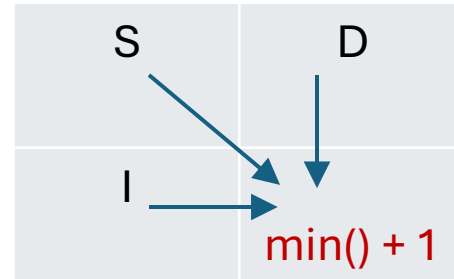
4 edits

F	O	O		D
M	O	N	E	Y
S		S	I	S

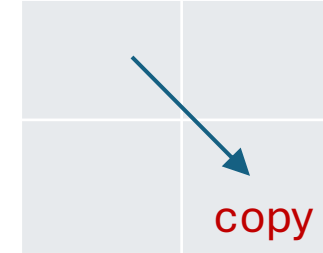
4 edits

# Example

if  $A[i] \neq B[j]$



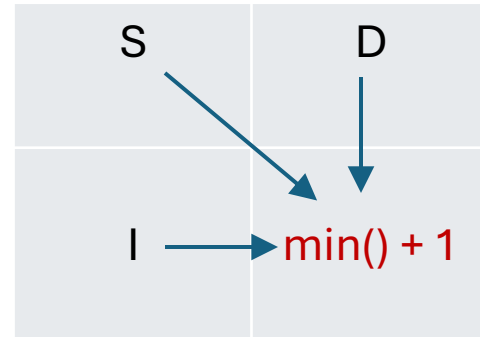
if  $A[i] = B[j]$



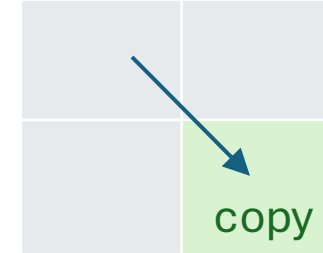
		B					
		M	O	N	E	Y	
		0	1	2	3	4	5
A	F	1					
	O	2					
	O	3					
	D	4					

# Example

if  $A[i] \neq B[j]$



if  $A[i] = B[j]$



		M	O	N	E	Y
	0	1	2	3	4	5
F	1	1	2	3	4	5
O	2	2	1	2	3	4
O	3	3	2	2	3	4
D	4	4	3	3	3	4

Diagram illustrating the dynamic programming table for the edit distance between "FOOD" and "MONY". The table shows the minimum number of operations (insertions, deletions, substitutions) required to transform one string into the other. The path from (0,0) to (4,6) is highlighted with blue arrows, showing the sequence of operations: Insert 'M', Insert 'O', Delete 'O', Delete 'O', and Delete 'D'. The final value 4 is circled in green.

# Recursion Tree

