# Design and Analysis of Algorithms Part II: Dynamic Programming

**Lecture 14: Minimum Edit Distance** 

盛浩

shenghao@buaa.edu.cn

北京航空航天大学计算机学院

北航《算法设计与分析》

#### 动态规划篇概述



- 在算法课程第二部分"动态规划"主题中,我们将主要聚焦于如下 经典问题:
  - 0-1 Knapsack (0-1背包问题)
  - Maximum Contiguous Subarray II (最大连续子数组 II)
  - Longest Common Subsequences (最长公共子序列)
  - Longest Common Substrings (最长公共子串)
  - Minimum Edit Distance (最小编辑距离)
  - Rod-Cutting (钢条切割)
  - Chain Matrix Multiplication (矩阵链乘法)

#### 动态规划篇概述

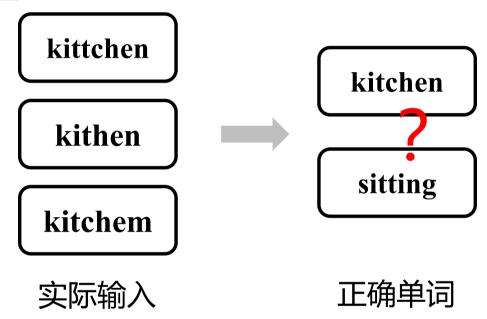


- 在算法课程第二部分"动态规划"主题中,我们将主要聚焦于如下 经典问题:
  - 0-1 Knapsack (0-1背包问题)
  - Maximum Contiguous Subarray II (最大连续子数组 II)
  - Longest Common Subsequences (最长公共子序列)
  - Longest Common Substrings (最长公共子串)
  - Minimum Edit Distance (最小编辑距离)
  - Rod-Cutting (钢条切割)
  - Chain Matrix Multiplication (矩阵链乘法)

# 问题背景



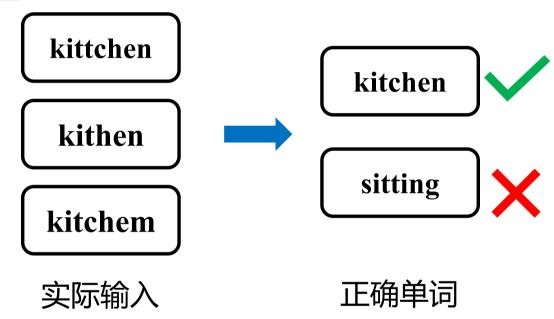
• 输入法自动更正



# 问题背景



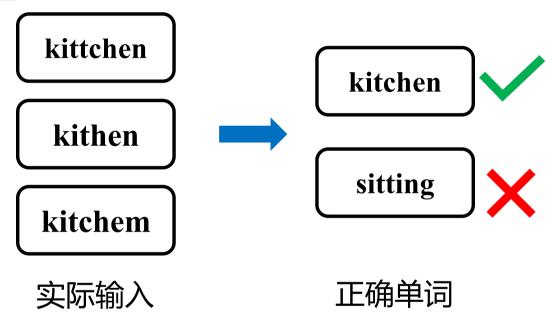
• 输入法自动更正



# 问题背景



• 输入法自动更正



问题:如何衡量序列的相似程度?



• 基本思想

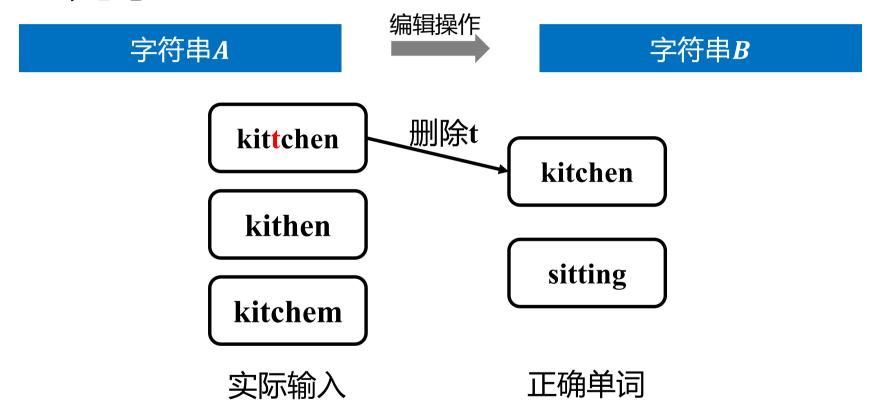
字符串A

编辑操作

字符串B

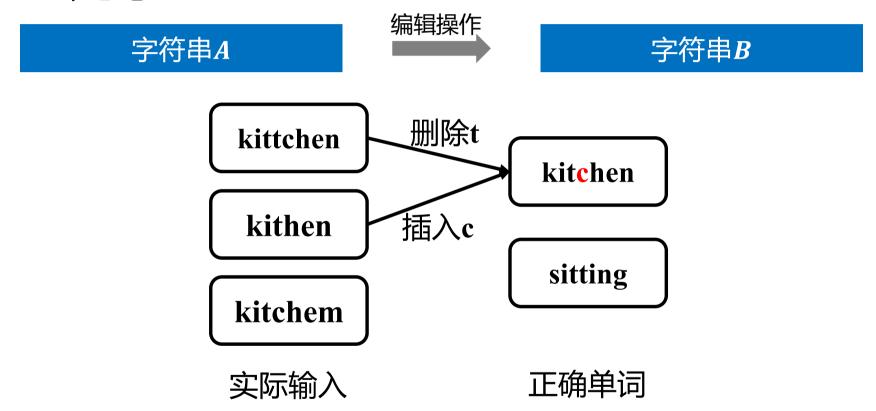


• 基本思想



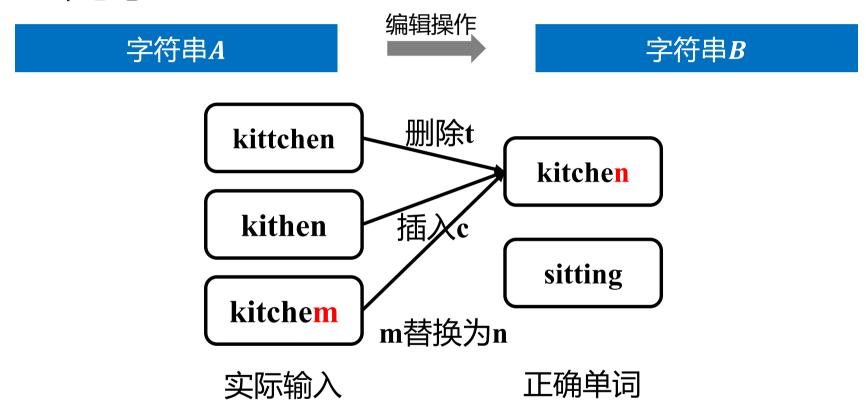


• 基本思想



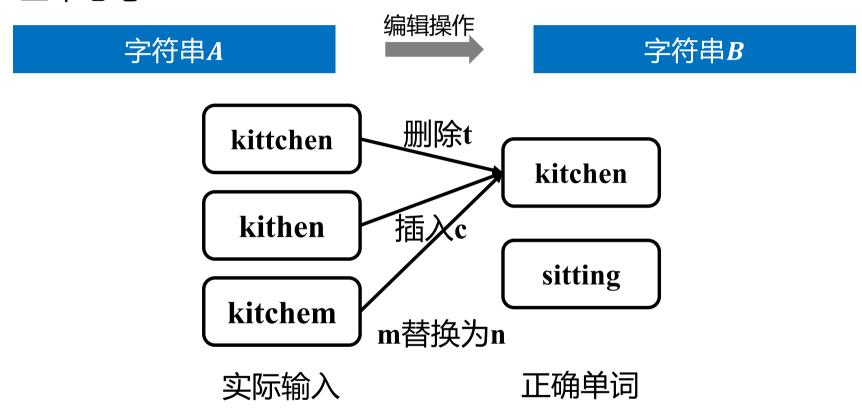


• 基本思想



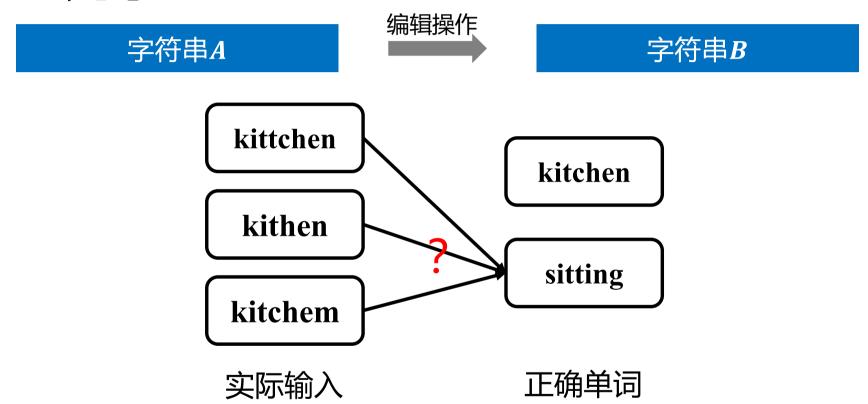


• 基本思想

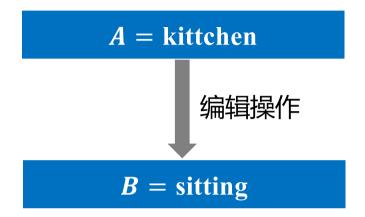




• 基本思想

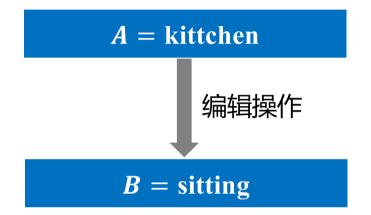






操作名称	操作示例
删除	kittchen → kitchen
插入	kithen → kit <b>c</b> hen
替换	kitchem → kitchen

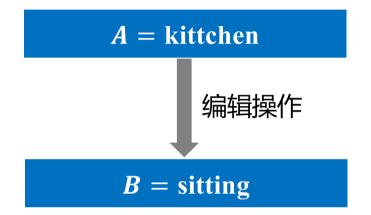




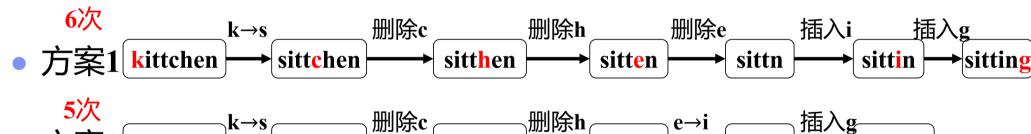
操作示例	操作名称
kittchen → kitchen	删除
kithen → kit <b>c</b> hen	插入
kitchem → kitche <mark>n</mark>	替换





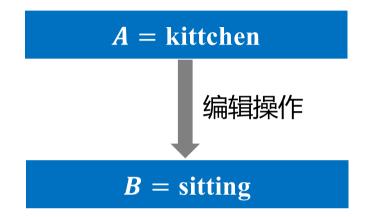


操作示例	操作名称
kittchen → kitchen	删除
kithen → kitchen	插入
kitchem → kitche <mark>n</mark>	替换

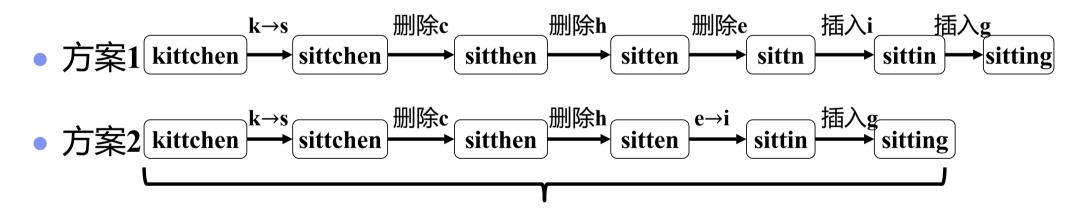


方案2 kittchen k→s sittchen 删除c sitthen 删除h sitten e→i sittin 插入g sitting



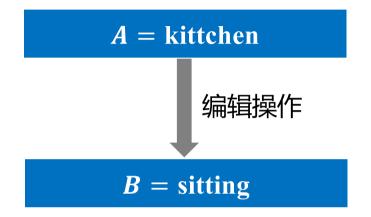


操作名称	操作示例
删除	kittchen → kitchen
插入	kithen → kit <b>c</b> hen
替换	kitchem → kitchen

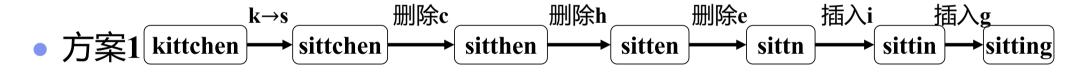


编辑距离:编辑操作次数





操作名称	操作示例
删除	kittchen → kitchen
插入	kithen → kit <b>c</b> hen
替换	kitchem → kitchen





问题:如何求出最少的编辑操作数(最小编辑距离)?



#### 编辑距离问题

#### Minimum Edit Distance, MED

#### 输入

- 长度为n的字符串s,长度为m的字符串t
- 输出
- 求出一组编辑操作 $0 = \langle e_1, e_2, ... e_d \rangle$ , 令

优化目标

min |0|

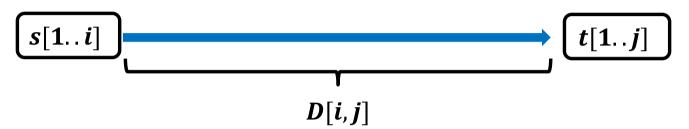
s.t. 字符串s经过o的操作后满足s=t

约束条件

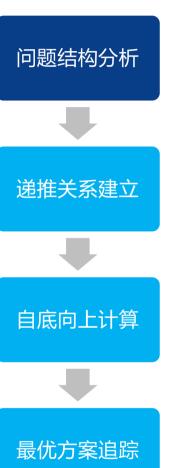
#### 问题结构分析



- 给出问题表示
  - D[i,j]:字符串S[1..i]变为t[1..j]的最小编辑距离



- 明确原始问题
  - D[n,m]:字符串s[1..n]变为t[1..m]的最小编辑距离



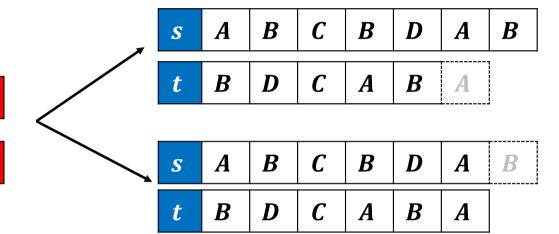
#### 递推关系建立:回顾与启发



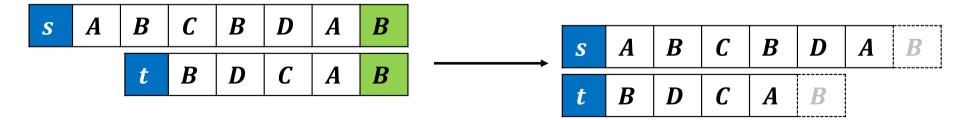
- 最长公共子序列
  - 如果 $s_i \neq t_j$

 s
 A
 B
 C
 B
 D
 A
 B

 t
 B
 D
 C
 A
 B
 A



• 如果 $s_i = t_i$ 

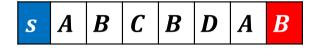


考察末尾元素

#### 递推关系建立

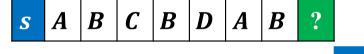


- 考察末尾元素
  - ●删除



t B D C A B A

插入



 $t \mid B \mid D \mid C \mid A \mid B \mid A$ 

替换



t B D C A B A

问题结构分析



递推关系建立

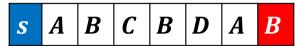


自底向上计算





• 考察末尾元素:删除





S A B C B D A B



问题结构分析



递推关系建立

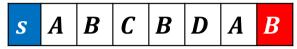


自底向上计算

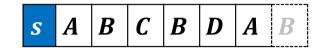




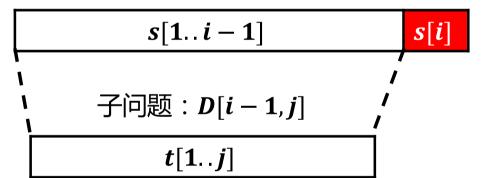
• 考察末尾元素:删除



 $t \mid B \mid D \mid C \mid A \mid B \mid A$ 



 $t \mid B \mid D \mid C \mid A \mid B \mid A$ 



问题结构分析



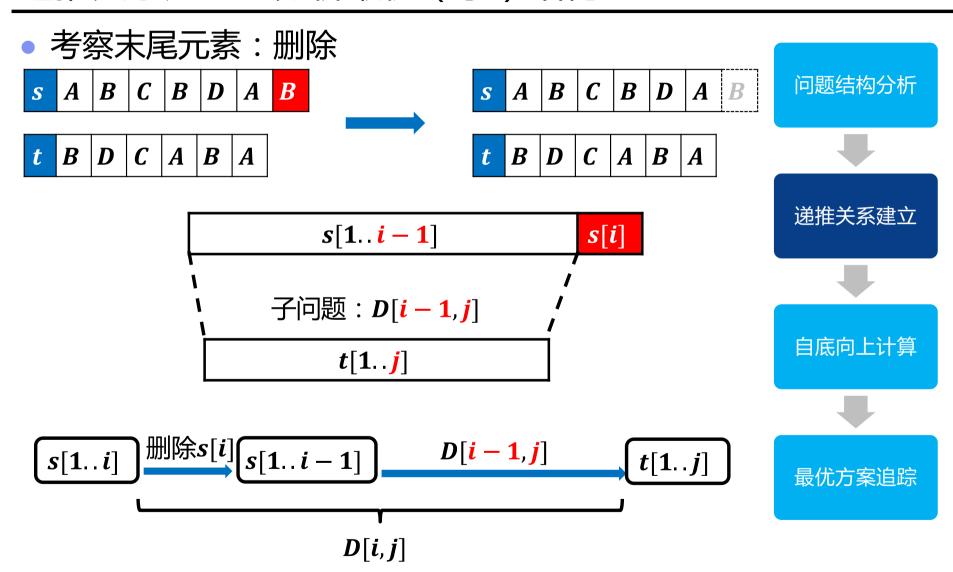
递推关系建立



自底向上计算

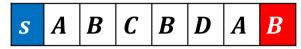




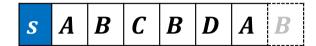




• 考察末尾元素:删除



 $t \mid B \mid D \mid C \mid A \mid B \mid A$ 



t B D C A B A



• D[i,j] = D[i-1,j] + 1

问题结构分析



递推关系建立

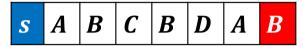


自底向上计算

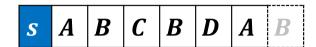




• 考察末尾元素:删除



 $t \mid B \mid D \mid C \mid A \mid B \mid A$ 



 $t \mid B \mid D \mid C \mid A \mid B \mid A$ 

问题结构分析



递推关系建立



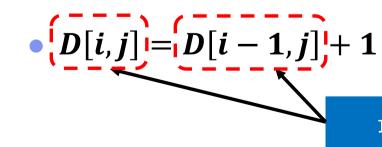
自底向上计算



最优方案追踪



最优子结构





• 考察末尾元素:插入

s A B C B D A B

t B D C A B A

问题结构分析

递推关系建立



自底向上计算





• 考察末尾元素:插入  $\boldsymbol{B}$ |C|B|D $\boldsymbol{B}$ 

 $t \mid B \mid D \mid C \mid$  $A \mid B$ 

|A|

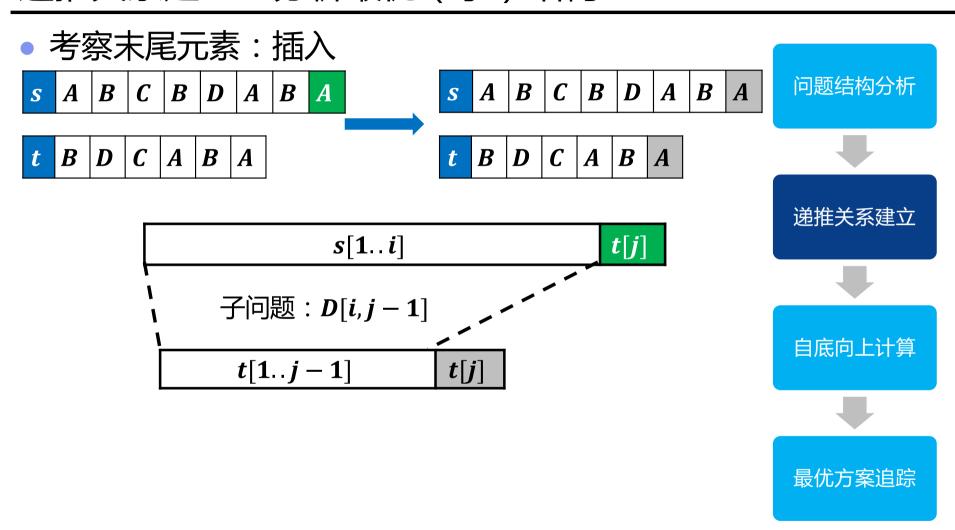
S A B $\boldsymbol{\mathcal{C}}$  $B \mid D$ |B||A|A

B |D| $C \mid A \mid B \mid A$  问题结构分析

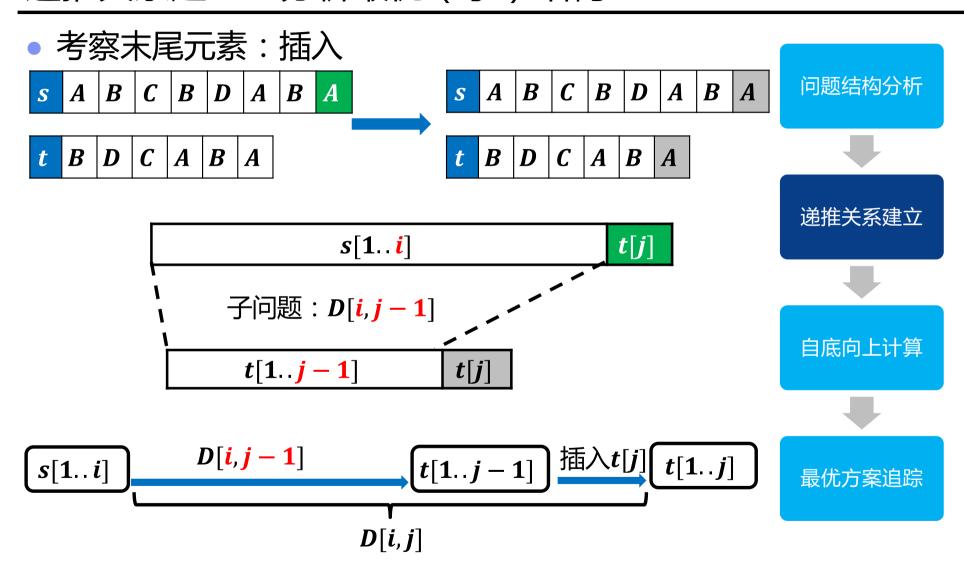
递推关系建立

自底向上计算

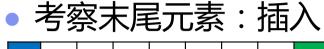


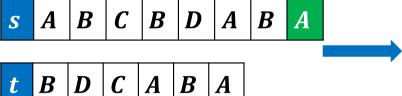




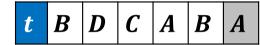


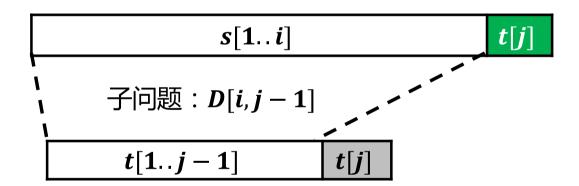






S A B C B D A B A





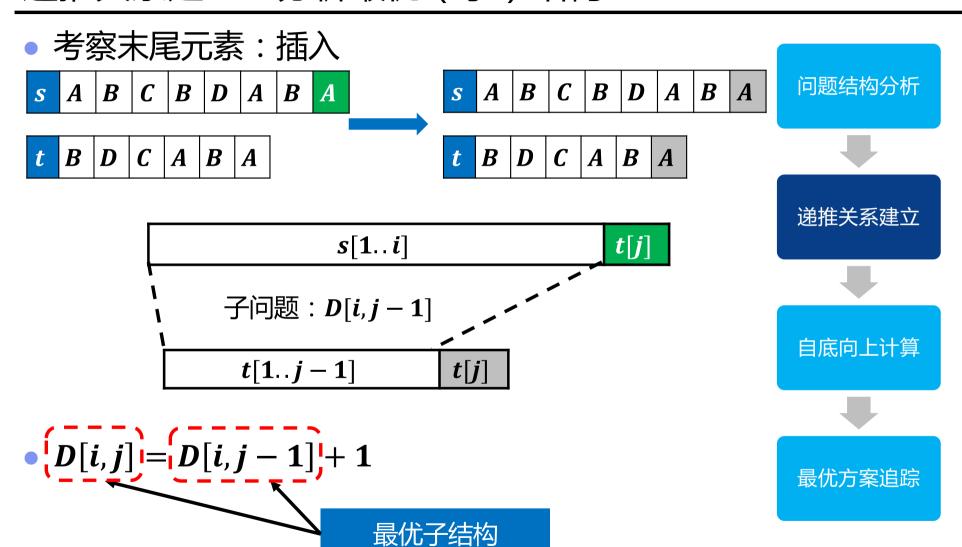
• D[i,j] = D[i,j-1] + 1

问题结构分析

递推关系建立

自底向上计算







• 考察末尾元素:替换

S A B C B D A B

 问题结构分析

递推关系建立

自底向上计算



• 考察末尾元素:替换

s A B C B D A ?

t B D C A B A

问题结构分析

递推关系建立

自底向上计算

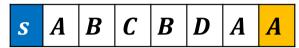
s[1..i-1]

子问题:**D**[i-1,j-1]

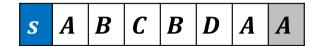
t[1..j-1]



• 考察末尾元素:替换



t B D C A B A



 $t \mid B \mid D \mid C \mid A \mid B \mid A$ 

s[i]

t[j]





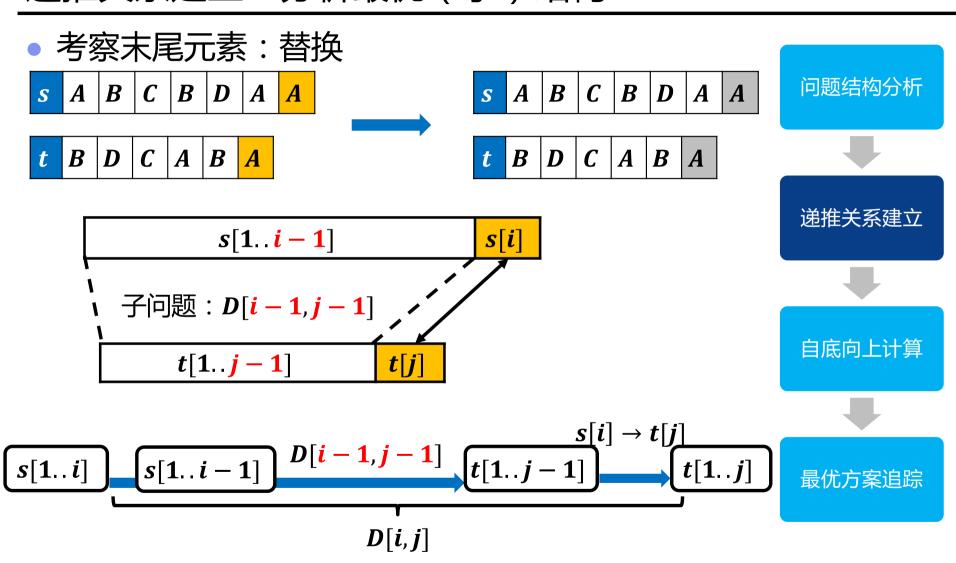
递推关系建立



自底向上计算



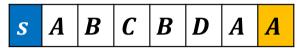




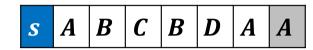
# 递推关系建立:分析最优(子)结构



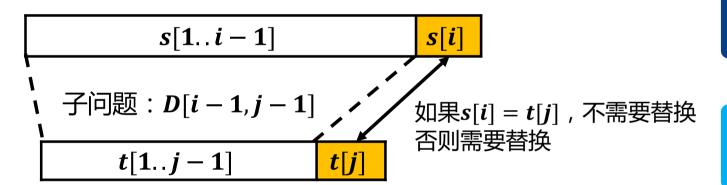
• 考察末尾元素:替换



t B D C A B A



t B D C A B A



•  $D[i,j] = D[i-1,j-1] + \begin{cases} 0, & \text{if } s[i] = t[j] \\ 1, & \text{if } s[i] \neq t[j] \end{cases}$ 

问题结构分析

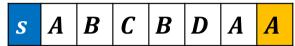


自底向上计算

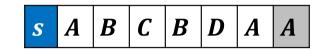
## 递推关系建立:分析最优(子)结构



• 考察末尾元素:替换



t B D C A B A



t B D C A B A

#### 问题结构分析



递推关系建立



自底向上计算



最优方案追踪

s[1..i-1]

一川迎: $D[\iota-1,J-1]$  如果s[i]=t[j],不需要替换 否则需要替换

s[i]

$$t[1..j-1] t[j]$$

最优子结构

# 递推关系建立:构造递推公式



• 综合上面三种方式

• 
$$D[i,j] = \min egin{cases} D[i-1,j]+1 & ext{删除} \ D[i,j-1]+1 & ext{插入} \ D[i-1,j-1]+egin{cases} 0, \ if \ s[i]=t[j] \ 1, \ if \ s[i] 
eq t[j] \end{cases}$$

问题结构分析



递推关系建立



自底向上计算



# 递推关系建立:构造递推公式



• 最小编辑距离 vs. 最长公共子序列

• 
$$D[i,j] = \min egin{cases} D[i-1,j]+1 & ext{删除} \ D[i,j-1]+1 & ext{插入} \ D[i-1,j-1]+egin{cases} 0, \ if \ s[i]=t[j] \ 1, \ if \ s[i] 
eq t[j] \end{cases}$$

•  $C[i,j] = \begin{cases} \max\{C[i-1,j], C[i,j-1]\}, x_i \neq y_j \\ C[i-1,j-1] + 1, x_i = y_j \end{cases}$ 

问题结构分析



递推关系建立



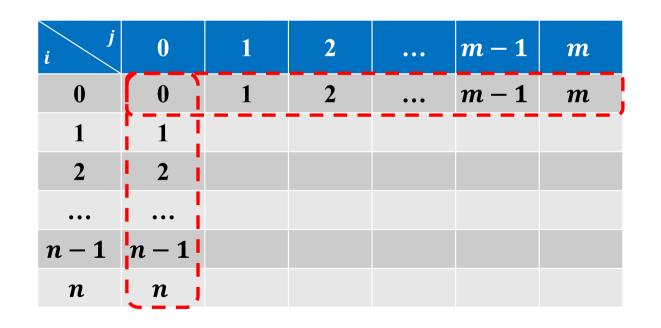
自底向上计算



# 自底向上计算:确定计算顺序



- 初始化
  - D[i,0]=i
    - 。 把长度为*i*的串变为空串至少需要*i*次操作(删除)
  - D[0,j]=j
    - 。 把空串变为长度为j的串至少需要j次操作(插入)



问题结构分析



递推关系建立



自底向上计算



# 自底向上计算:确定计算顺序



#### • 递推公式

i	0	1	2	•••	m-1	m
0	0	1	2	•••	m-1	m
1	1	0[i-1,i]	<b>-1</b> ]+{	0	_	
2	2	[c =, j	-1 (	$\begin{array}{c} 1 & D[i] \\ \cdot & \downarrow \end{array}$	-1, j] +	1
• • •	•••			D[i,j]		
n-1	n-1		D[i,j-1]	1]+1		
$\boldsymbol{n}$	$\boldsymbol{n}$					

问题结构分析



递推关系建立



自底向上计算



# 自底向上计算:依次计算问题



#### • 递推公式

• 
$$D[i,j] = \min egin{cases} D[i-1,j]+1 & ext{删除} \ D[i,j-1]+1 & ext{插入} \ D[i-1,j-1]+iggl\{0,\ if\ s[i]=t[j] \ 1,\ if\ s[i] 
otation iggl\} iggl\}$$

i j	0	1	2	•••	m-1	m
0	0	1	2	•••	m-1	m
1	1					<del></del> >
2	2	<u> </u>				<del></del> >
• • •	•••	4				<del></del> >
n-1	n-1	4			_====	<del>.=&gt;</del>
n	n	<b>4</b>			<b></b>	*

问题结构分析



递推关系建立



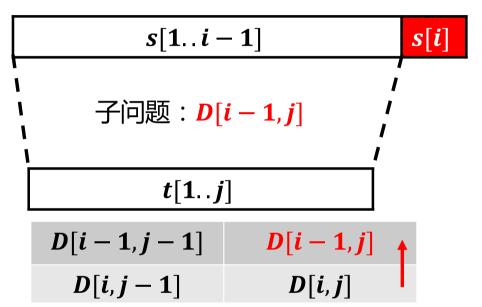
自底向上计算



# 最优方案追踪:记录决策过程



• 追踪数组Rec,记录子问题来源



Rec[i,j]	子问题来源	操作
U	上侧,即 <b>D</b> [i-1,j]	删除s[i]

问题结构分析



递推关系建立



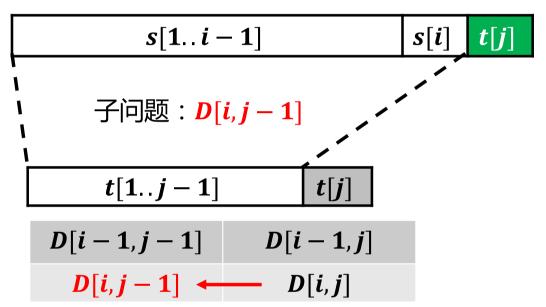
自底向上计算



# 最优方案追踪:记录决策过程



• 追踪数组Rec,记录子问题来源



Rec[i,j]	子问题来源	操作
U	上侧,即 <b>D</b> [i-1,j]	删除s[i]
L	左侧,即 <b>D[i,j-1</b> ]	插入 <b>t</b> [ <b>j</b> ]

问题结构分析



递推关系建立



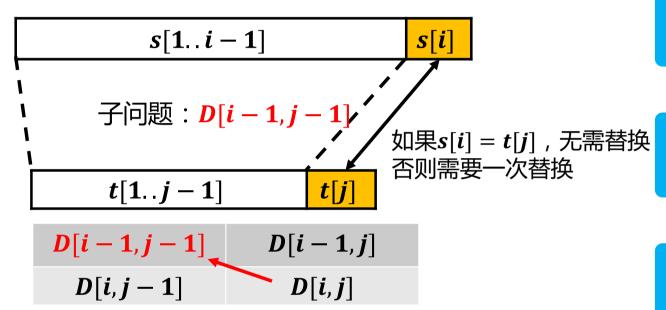
自底向上计算



## 最优方案追踪:记录决策过程



• 追踪数组Rec,记录子问题来源



Rec[i,j]	子问题来源	操作		
U	上侧,即 <b>D</b> [i-1,j]	删除s[i]		
L	左侧,即 <b>D</b> [i,j-1]	插入 <b>t</b> [ <b>j</b> ]		
LU	左上,即 <b>D</b> [i-1,j-1]	用 $t[j]$ 替换 $s[i]$ /无操作		

问题结构分析



递推关系建立

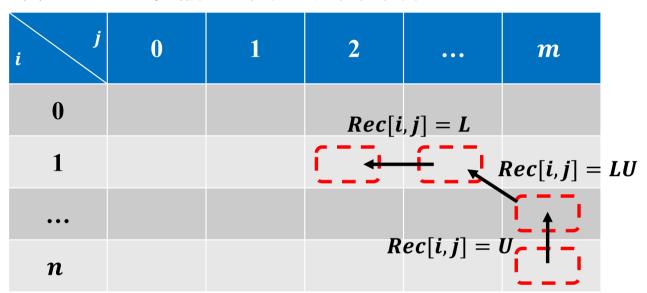




# 最优方案追踪:输出最优方案



• 根据数组Rec,输出最少编辑操作



Rec[i,j]	子问题来源	操作
U	上侧,即 <b>D</b> [i-1,j]	删除s[i]
L	左侧,即 <b>D</b> [i,j-1]	插入 <i>t</i> [ <i>j</i> ]
LU	左上,即 <b>D</b> [i-1,j-1]	用 $t[j]$ 替换 $s[i]$ /无操作

问题结构分析



递推关系建立



自底向上计算





	1	2	3	4	5	6	7
S	A	В	C	В	D	A	В
t	В	D	C	A	В	A	

D								Rec							
i $j$	0	1	2	3	4	5	6	i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6	0		L	L	L	L	L	L
1	1							1	U						
2	2		初始	· 北.				2	U						
3	3			110				3	U						
4	4							4	U						
5	5							5	U						
6	6							6	U						
7	7							7	U						

 $\boldsymbol{D}$ 



	1	2	3	4	5	6	7
S	_A_	В	C	В	D	A	В
t	_B_	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1						
2	2						
3	3						
4	4						
5	5						
6	6						
7	7						

Rec							
i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U						
2	U						
3	U						
4	U						
5	U						
6	U						
7	U						



	1	2	3	4	5	6	7		D[i-1,j]+1
S	_A_}	В	C	В	D	A	В	$D[i,j] = \min \langle$	D[i,j-1]+1 $(0,if,s[i]-t[i])$
t	В	D	C	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$
D		I	O[i-1]	<i>j</i> ]			Rec		

D			$\underline{\hspace{0.1cm}}^{D[\iota]}$	-1,J				Rec							
i $j$	0	1	2	3	4	5	6	i	0	1	2	3	4	5	6
0	0	1_	2	3	4	5	6	0		L	L	L	L	L	L
1	1							1	U						
2	2							2	U						
3	3							3	U						
4	4							4	U						
5	5							5	U						
6	6							6	U						
7	7							7	U						



删除A

D[i-1,j]+1

		_		_		V	<u>'</u>				ם <b>ת</b>		1.4	插入B
S	A	В	C	В	D	A	В	D[i,j]	= min	$\{$				
t	В	D	C	A	В	A				D[i -	-1, j-1	$oldsymbol{1}$ ] + $ig\{oldsymbol{1}$	, if s[i] , if s[i]	$= t_{[j]}$ $\neq t[j]$
D			O[i-1,	<i>j</i> ]			Rec							
i	0	1 2	2 3	4	5	6	i $j$	0	1	2	3	4	5	6
0	0	1 2	2 3	4	5	6	0		L	${f L}$	L	L	L	L
1	1	;					1	U						
2	2			ш,			2	U						
3	3	D	[i, j-1]				3	U						
4	4						4	U						
5	5						5	U						
6	6						6	U						
7	7						7	U						



S	1 A	2 B		3 C	4 B	5 D	6 A	7 B	<b>D</b> [i, j]	= min	\{ \begin{align*} \begin{align*} \langle \begin{align*} \langle \begin{align*} \langle \langle \end{align*} \rangle \langle \langle \end{align*}	D[i]	i-1,j] $i,j-1$ $i,j-1$	+ 1	删除A 插入B = $t[j]$
t	_B	, D		$\mathbf{C}^{\mathbf{S}[1]}$	$t] \neq t[$		A				D[l-	- <b>1</b> , <b>j</b> – 1	1) + (1	, if $s[i]$	$\neq t[j]$
$\boldsymbol{D}$			D[i]	-1,j	]			Rec					A春	替换为I	3
i	0	1	2	3	4	5	6	i	0	1	2	3	4	5	6
0	0_	1	2	3	4	5	6	0		L	L	L	L	L	L
1	(1)			D	[i-1,	[j-1]		1	U						
2	2							2	U						
3	3		D[i,]	[j-1]				3	U						
4	4							4	U						
5	5							5	U						
6	6							6	U						
7	7							7	U						



	1	2		3	4	5	6	7					[i-1,j]		
S	A	B		<b>C</b>	В	D	A	В	D[i,j]	= min	{		[i, j-1]		_ +[i]
t	В	D		$\mathbf{C}^{s[}$	$i] \neq t[$	<i>j</i> ]	A				D[i -	-1, j-1	$oldsymbol{1}$ ] + $\left\{egin{matrix} oldsymbol{0} \\ oldsymbol{1} \end{array}\right\}$	, if s[i] , if s[i]	$= t[j]$ $\neq t[j]$
D			D[i]	<b>– 1</b> , j	7]			Rec			·				
i	0	1	2	3	4	5	6	i	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6	0		L	L	L	L	L	L
1	1	1		D	i-1	(j-1]		1	U	LU					
2	2							2	U						
3	3		D[i, ]	j-1				3	U						
4	4							4	U						
5	5							5	U						
6	6							6	U						
7	7							7	U						



	1	2	3	4	5	6	7					[i-1,j]		
S	A	В	C	В	D	A	В	D[i,j]	= min	. {		i, j — 1] (ຄ		=t[i]
t	В	D	, C	S	$i] \neq t[$	j]				D[i -	- 1, <i>j</i> –	$1$ ] + $\left\{1\right\}$	, if s[i] , if s[i]	$\neq t[j]$
D				D[i-1]	., <b>j</b> ]		Rec			`				
i	0	1	2 3	4	5	6	i	0	1	2	3	4	5	6
0	0	1	2 3	4	5	6	0		L	L	L	L	L	L
1	1	(1)	2		0[i-1]	[,j-1]	1	U	LU	LU	)			
2	2						2	U						
3	3		D	0[i,j-1]	.]		3	U						
4	4						4	U						
5	5						5	U						
6	6						6	U						
7	7						7	U						

 $\boldsymbol{D}$ 



	1	2	3	4	5	6	7
S	A	В	C	В	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3			
2	2						
3	3						
4	4						
5	5						
6	6						
7	7						

i	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU			
2	U						
3	U						
4	U						
5	U						
6	U						
7	U						



s t	1 (_A, B	2 3 E	3	3 C C	4 B		$\begin{bmatrix} 6 \\ A \\ [i] = t \\ i - 1, \end{bmatrix}$		D[i,j]	= min	$\left\{ D[i -$	D[	i — 1, j] i, j — 1] 1] + { <mark>0</mark> 1] 无持	+ 1	
i $j$	0	1	2	3	4	5	6	i	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6	0		L	L	L	L	L	L
1	1	1	2	3	3		D[	i-1,j	<b>– 1</b> ]	LU	LU	LU	LU		
2	2							2	U						
3	3					D[i,	j – 1]	3	U						
4	4							4	U						
5	5							5	U						
6	6							6	U						
7	7							7	U						



	1	2	3	4	5	6	7
S	A	В	C	В	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

D							
i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	
2	2						
3	3						
4	4						
5	5						
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	
2	U						
3	U						
4	U						
5	U						
6	U						
7	U						

D

6

7

6



	1	2	3	4	5	6	7
S	A	В	C	В	D	A	В
t	В	D	C	A	В	A	

 $D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$ 

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5_	6_
1	1	1	2	3	3	4	5
2	2						
3	3						
4	4						
5	5						
_	_						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U						
3	U						
4	U						
5	U						
6	U						
7	U						

 $\boldsymbol{D}$ 



	1	2	3	4	5	6	7
S	A	B	C	В	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1					
3	3						
4	4						
5	5						
6	6						
7	7						

Rec							
i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU					
3	U						
4	U						
5	U						
6	U						
7	U						

 $\boldsymbol{D}$ 



	1	2	3	4	5	6	7
S	A	B	C	В	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	[1	2	j			
3	3						
4	4						
5	5						
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU				
3	U						
4	U						
5	U						
6	U						
7	U						

 $\boldsymbol{D}$ 



	1	2	3	4	5	6	7
S	A	B	C	В	D	A	В
t	В	D	$\begin{bmatrix} \mathbf{C} \end{bmatrix}$	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	j		
3	3						
4	4						
5	5						
6	6						
7	7						

i	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU			
3	U						
4	U						
5	U						
6	U						
7	U						



	1	2	3	4	5	6	7
S	A	B	C	В	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

D							
i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4		
3	3						
4	4						
5	5						
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU		
3	U						
4	U						
5	U						
6	U						
7	U						



	1	2	3	4	5	6	7
S	A	B	C	В	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

D							
i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	
3	3						
4	4						
5	5						
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	
3	U						
4	U						
5	U						
6	U						
7	U						



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	B	C	В	D	A	В	$D[i,j] = \min \langle$	D[i,j-1]+1 $(0 if s[i] = t[i]$
t	В	D	C	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

D

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3						
4	4						
5	5						
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U						
4	U						
5	U						
6	U						
7	U						

 $\boldsymbol{D}$ 



	1	2	3	4	5	6	7
S	A	В	C	В	D	A	В
t	<u>B</u>	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

0	1	2	3	4	5	6
0	1	2	3	4	5	6
1	1	2	3	3	4	5
2	$\begin{bmatrix} 1 \end{bmatrix}$	2	3	4	3	4
3	2					
4						
5						
6						
7						
	0 1 2 3 4 5 6	0 1 1 1 2 1 3 2 4 5 6	0     1     2       1     1     2       2     1     2       3     2     4       5     6	0       1       2       3         1       1       2       3         2       1       2       3         3       2       4       5         6       6       6       6	0       1       2       3       4         1       1       2       3       3         2       1       2       3       4         3       2       4       5         6       6       6       6	0     1     2     3     4     5       1     1     2     3     3     4       2     1     2     3     4     3       3     2     3     4     3       4     4     4     4     4       5     6     6     6

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U					
4	U						
5	U						
6	U						
7	U						

 $\boldsymbol{D}$ 



	1	2	3	4	5	6	7
S	A	В	C	В	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2				
4	4						
5	5						
6	6						
7	7						

Rec							
i $j$	0	1	2	3	4	5	6
0		L	${f L}$	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU				
4	U						
5	U						
6	U						
7	U						



	1	2	3	4	5	6	7
S	A	В	C	В	D	A	В
t	В	D	C	A	В	A	

 $D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$ 

D							
i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2			
4	4						
5	5						
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU			
4	U						
5	U						
6	U						
7	U						



	1	2	3	4	5	6	7	
S	A	В	C	В	D	A	В	L
t	В	D	C	A	В	A		

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

D							
i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3		
4	4						
5	5						
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L		
4	U						
5	U						
6	U						
7	U						



	1	2	3	4	5	6	7	
S	A	В	(C)	В	D	A	В	D[i,
t	В	D	C	A	B	A		

 $D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t \end{cases}$ 

•		-		
- /	,		ı	
- 4	,	1	,	

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	
4	4						
5	5						
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	
4	U						
5	U						
6	U						
7	U						



	1	2	3	4	5	6	7
S	A	В	(C)	В	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t \end{bmatrix}$$

D	

i j	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4						
5	5						
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U						
5	U						
6	U						
7	U						

 $\boldsymbol{D}$ 



	1	2	3	4	5	6	7
S	A	В	C	B	D	A	В
t	B	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3					
5	5						
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU					
5	U						
6	U						
7	U						

 $\boldsymbol{D}$ 



	1	2	3	4	5	6	7
S	A	В	C	В	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3				
5	5						
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU				
5	U						
6	U						
7	U						

 $\boldsymbol{D}$ 



	1	2	3	4	5	6	7
S	A	В	C	B	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3			
5	5						
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU			
5	U						
6	U						
7	U						

 $\boldsymbol{D}$ 



	1	2	3	4	5	6	7
S	A	В	C	B	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3		3		
5	5						
6	6						
7	7						

i	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU		
5	U						
6	U						
7	U						

D



L

LU

L

LU

	1	2	3	4	5	6	7	
S	A	В	C	B	D	A	В	D[i]
t	В	D	C	A	B	A		

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t \end{cases}$$

i $j$	0	1	2	3	4	5	6
		1					
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4

i $j$	0	1	2	3	4	5
0		L	L	L	L	L
1	U	LU	LU	LU	LU	L
2	U	LU	LU	LU	LU	LU
3	U	U	LU	LU	L	L
4	U	LU	LU	LU	LU	LU
5	U					
6	U					
7	U	_		_		

D

7



	1	2	3	4	5	6	7
S	A	В	C	B	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

i	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5						
6	6						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U						
6	U						
7	U						

 $\boldsymbol{D}$ 



	1	2	3	4	5	6	7	
S	A	В	C	В	D	A	В	L
t	В	D	C	A	В	A		

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

i j	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3				
6	6						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU				
6	U						
7	U						



	1	2	3	4	5	6	7	
S	A	В	C	В	D	A	В	D[i,j] =
t	В	D	C	A	В	A		

 $D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t \end{bmatrix}$ 

D
---

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	j		
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU			
6	U						
7	U						

 $\boldsymbol{D}$ 



	1	2	3	4	5	6	7	
S	A	В	C	В	D	A	В	D[i,j]
t	В	D	C	A	В	A		

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t \end{cases}$$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	4		
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU		
6	U						
7	U						

D



	1	2	3	4	5	6	7	
S	A	В	C	В	D	A	В	D[i,j]
t	В	D	C	A	B	A		

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t \end{cases}$$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	4	4	
6	6						
7	7						

i	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	${f L}$	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	${f L}$	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	
6	U						
7	U						



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	В	C	В	D	A	В	$D[i,j] = \min \langle$	D[i, j-1] + 1 $(0 if s[i] = t[i]$
t	В	D	C	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

 $\boldsymbol{D}$ 

i j	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	4	4	4
6	6						
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U						
7	U						

7



	1	2	3	4	5	6	7		D[i-1,j]+1
S			C	В	D	A	В	$D[i,j] = \min \langle$	D[i, j-1] + 1 $(0 if s[i] = t[i]$
t	В	D	C	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

 0
 0
 1
 2
 3
 4
 5
 6

 1
 1
 1
 2
 3
 3
 4
 5

2 2 1 2 3 4 3 4

 3
 3
 2
 2
 2
 3
 4
 4

 4
 4
 3
 3
 3
 3
 3
 4

5 <u>5</u> <u>4</u> 3 4 4 4 4 6 6 5

i	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U					
7	U						



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	В	C	В	D	A	В	$D[i,j] = \min \langle$	D[i, j-1] + 1 $(0 if s[i] = t[i]$
t	В	D	C	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

i j	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	4	4	4
6	6	5	4				
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U				
7	U						



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	В	C	В	D	A	В	$D[i,j] = \min \langle$	D[i,j-1]+1 $(0 if s[i] = t[i]$
t	В	D	C	A	В	A			$\begin{cases} D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases} \end{cases}$

i j	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	4	4	4
6	6	5	4	4			
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU			
7	U						



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	В	C	В	D	A	В	$D[i,j] = \min \langle$	D[i, j-1] + 1 $(0 if s[i] = t[i]$
t	В	D	C	A	В	A			$\begin{cases} D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases} \end{cases}$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	4	4	4
6	6	5	4	4	4		
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU		
7	U						



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	В	C	В	D	A	В	$D[i,j] = \min \langle$	D[i, j-1] + 1 $(0 if s[i] = t[i]$
t	В	D	C	A	B	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

i	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	4	4	4
6	6	5	4	4	4	5	
7	7						

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	
7	U						



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	В	C	В	D	A	В	$D[i,j] = \min \langle$	D[i, j-1] + 1 $(0 if s[i] = t[i]$
t	В	D	C	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

 $\boldsymbol{D}$ Rec 3 3 2 5 6 0 5 0 4 4 6 0 0 2 3 5 0 L L L 4 L L 6 3 2 3 5 4 1 U LU LU LU LU L LU 3 LU LU LU LU LU 2 2 3 U L 4 4 LU 3 3 2 2 2 3 3 U U LU L LU 4 L 4 3 3 3 3 3 LU LU LU LU 4 4 U 4 4 LU L 5 5 3 4 5 U U LU LU LU LU LU 4 4 LU LU 5 LU LU 6 6 4 4 4 6 U U U 7 7 7 U



	1	2	3				7		D[i-1,j]+1
S	A	В	C	В	D	A	B	$D[i,j] = \min \langle$	D[i,j-1]+1 $(0 if s[i] = t[i]$
t	B	D	C	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

i j	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	4	4	4
6	6	5	4	4	4	5	4
7	7	6					

i	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU					



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	В	C	В	D	A	B	$D[i,j] = \min \langle$	D[i, j-1] + 1 $(0, if s[i] = t[i]$
t	В	D	C	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

D

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	4	4	4
6	6	5	4	4	4	5	4
7	7	6	5				

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U				



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	В	C	В	D	A	B	$D[i,j] = \min \langle$	D[i, j-1] + 1 $(0 if s[i] = t[i]$
t	В	D	C	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	4	4	4
6	6	5	4	4	4	5	4
7	7	6	5	5			

i	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U	LU			



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	В	C	В	D	A	B	$D[i,j] = \min \langle$	D[i,j-1]+1 $(0 if s[i]-t[i]$
t	В	D	C	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	4	4	4
6	6	5	4	4	4	5	4
7	7	6	5	5	5		

i	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U	LU	LU		



	1						7		D[i-1,j]+1
S	A	В	C	В	D	A	BD	$O[i,j] = \min \langle$	D[i,j-1]+1 $(0 if s[i] = t[i]$
t	В	D	C	A	B	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

 $\boldsymbol{D}$ 

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	4	4	4
6	6	5	4	4	4	5	4
7	7	6	5	5	5	4	

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U	LU	LU	LU	



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	В	C	В	D	A	B	$D[i,j] = \min \langle$	D[i,j-1]+1 $(0 if s[i] = t[i]$
t	В	D	C	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

D

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3	4	4	4	4
6	6	5	4	4	4	5	4
7	7	6	5	5	5	4	5

i	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U	LU	LU	LU	L



	1	2	3	4	5	6	7
S	A	В	C	В	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

D

i $j$	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	1	2	3	3	4	5
2	2	1	2	3	4	3	4
3	3	2	2	2	3	4	4
4	4	3	3	3	3	3	4
5	5	4	3		最优制	罕	4
6	6	5	4	4	4	3	4
7	7	6	5	5	5	4	5

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U	LU	LU	LU	L



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	В	C	В	D	A	В	$D[i,j] = \min \langle$	D[i, j-1] + 1 $(0 if s[i] = t[i]$
t	В	D	C	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

Rec

操作:

插入A

i	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U	LU	LU	LU	L



	1	2	3	4	5	6	7	$ \left( D[i-1,j]+1\right) $
S	A	В	C	В	D	A	D[i,j] = mir	D[i,j-1]+1 $(0 if s[i] = t[i]$
t	В	D	C	A	В	A		$ \begin{cases} D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases} $

操作:

无需操作 插入A

i j	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U	LU	LU	LU	L



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	В	C	В	D	A	В	$D[i,j] = \min \langle$	D[i,j-1]+1 $(0 if s[i] = t[i]$
t	В	D	C	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

操作:

无需操作 无需操作 插入A

i j	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U	LU	LU	LU	L



	1	2	3	4	5	6	7		D[i-1,j]+1
S	A	В	C	В	D	A	В	$D[i,j] = \min \langle$	D[i,j-1]+1 $(0 if s[i] = t[i]$
t	В	D	$\mathbb{C}$	A	В	A			$D[i-1,j-1] + \begin{cases} 0, if \ s[i] = t[j] \\ 1, if \ s[i] \neq t[j] \end{cases}$

操作:

用C替换D 无需操作 无需操作 插入A

i	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U	LU	LU	LU	${f L}$



	1	2	3	4	5	6	7	
S	A	В	C	В	D	A	В	D
t	В	D	C	A	В	A		

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

操作:

用D替换B 用C替换D 无需操作 无需操作 插入A

i	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U	LU	LU	LU	L



	1	2	3	4	5	6	7		
S	A	В	C	В	D	A	В	$D[i,j] = \min \{$	}
t	В	D	C	A	В	A			

 $D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
eq t[j] \end{cases}$ 

操作:

删除C 用D替换B 用C替换D 无需操作 无需操作 插入A

i $j$	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U	LU	LU	LU	L



	1	2	3	4	5	6	7
S	A	B	C	В	D	A	В
t	В	D	C	A	В	A	

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t[j] \end{cases}$$

操作:

无需操作 删除C 用D替换B 用C替换D 无需操作 无需操作 插入A

i j	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U	LU	LU	LU	L



	1	2	3	4	5	6	7	
S	A	В	C	В	D	A	В	<b>D</b> [
t	В	D	C	A	В	A		

 $D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
otag t \end{cases}$ 

操作: 删除A 无需操作

删除C

用D替换B

用C替换D

无需操作

无需操作

插入A

i	0	1	2	3	4	5	6
0		L	L	L	L	L	L
1	U	LU	LU	LU	LU	L	LU
2	U	LU	LU	LU	LU	LU	L
3	U	U	LU	LU	L	L	LU
4	U	LU	LU	LU	LU	LU	L
5	U	U	LU	LU	LU	LU	LU
6	U	U	U	LU	LU	LU	LU
7	U	LU	U	LU	LU	LU	L



```
输入: 字符串s和t
 输出: s和t的最小编辑距离
n \leftarrow \text{length}(s)
m \leftarrow \text{length}(t)
 新建D[0..n,0..m],Rec[0..n,0..m]两个二维数组
 //初始化
for i \leftarrow 0 to n do
                                            初始化
     D[i,0] \leftarrow i
     Rec[i,0] \leftarrow ``U"
end
for j \leftarrow 0 to m do
     D[0,j] \leftarrow j
     Rec[0,j] \leftarrow ``L"
\operatorname{end}
```



```
//动态规划
for i \leftarrow 1 to n do
                                                     □依次计算子问题
     for j \leftarrow 1 to m do
          if s_i \neq t_i then
          c \leftarrow 1
          end
         replace \leftarrow D[i-1, j-1] + c
          delete \leftarrow D[i-1,j]+1
         insert \leftarrow D[i, j-1] + 1
         if replace = min\{delete, insert, replace\} then
             D[i,j] \leftarrow D[i-1,j-1] + c
             Rec[i,j] \leftarrow "LU"
          end
          else if insert = min\{delete, insert, replace\} then
             D[i,j] \leftarrow D[i,j-1] + 1
             Rec[i,j] \leftarrow "L"
          end
          else
             D[i,j] \leftarrow D[i-1,j] + 1
             Rec[i,j] \leftarrow ``U"
         end
     \mathbf{end}
  end
```



```
//动态规划
for i \leftarrow 1 to n do
    for j \leftarrow 1 to m do
                                                           替换/无需操作
        if s_i \neq t_i then
         \downarrow c \leftarrow 1
        replace \leftarrow D[i-1,j-1] + c
        delete \leftarrow D[i-1,j]+1
        insert \leftarrow D[i, j-1] + 1
        if replace = min\{delete, insert, replace\} then
            D[i,j] \leftarrow D[i-1,j-1] + c
             Rec[i,j] \leftarrow "LU"
        end
        else if insert = min\{delete, insert, replace\} then
             D[i,j] \leftarrow D[i,j-1] + 1
            Rec[i,j] \leftarrow "L"
        end
        else
             D[i,j] \leftarrow D[i-1,j] + 1
            Rec[i,j] \leftarrow "U"
        \mathbf{end}
    \mathbf{end}
\mathbf{end}
```



```
//动态规划
for i \leftarrow 1 to n do
     for j \leftarrow 1 to m do
          c \leftarrow 0
         if s_i \neq t_j then c \leftarrow 1
          end
         replace \leftarrow D[i-1, j-1] + c
          delete \leftarrow D[i-1,j]+1
        insert \leftarrow D[i, j-1] + 1
         \begin{array}{c} \textbf{if } replace = min\{delete, insert, replace\} \textbf{ then} \\ + \mathcal{D}[i,j] \leftarrow \mathcal{D}[i-1,j-1] + c \end{array} 
                                                                                      采用替换操作
              Rec[i,j] \leftarrow "LU"
          end
          else if insert = min\{delete, insert, replace\} then
              D[i,j] \leftarrow D[i,j-1] + 1
              Rec[i,j] \leftarrow "L"
          end
          else
              D[i,j] \leftarrow D[i-1,j] + 1
              Rec[i,j] \leftarrow "U"
          end
     end
end
```



```
//动态规划
for i \leftarrow 1 to n do
     for j \leftarrow 1 to m do
          c \leftarrow 0
          if s_i \neq t_j then c \leftarrow 1
          end
          replace \leftarrow D[i-1, j-1] + c
          delete \leftarrow D[i-1,j]+1
          insert \leftarrow D[i, j-1] + 1
          \begin{array}{c} \textbf{if } replace = min\{delete\_insert, replace\} \textbf{ then } \\ \mid D[i,j] \leftarrow D[i-1,j-1] + c \end{array} 
                                                                                               记录编辑距离和操作
              Rec[i,j] \leftarrow "LU"
          \mathbf{else} \ \mathbf{if} \ insert = min\{delete, insert, replace\} \ \mathbf{then}
                D[i,j] \leftarrow D[i,j-1] + 1
               Rec[i,j] \leftarrow "L"
          end
          else
                D[i,j] \leftarrow D[i-1,j] + 1
               Rec[i,j] \leftarrow ``U"
          end
     \mathbf{end}
\mathbf{end}
```



```
//动态规划
for i \leftarrow 1 to n do
    for j \leftarrow 1 to m do
         c \leftarrow 0
         if s_i \neq t_j then c \leftarrow 1
         end
         replace \leftarrow D[i-1, j-1] + c
         delete \leftarrow D[i-1,j]+1
         insert \leftarrow D[i, j-1] + 1
         if replace = min\{delete, insert, replace\} then
             D[i,j] \leftarrow D[i-1,j-1] + c
             Rec[i,j] \leftarrow ``LU"
         \mathbf{end}
       \mathbf{f} else if insert = min\{delete, insert, replace\} then
                                                                                      采取插入操作
             D[i,j] \leftarrow D[i,j-1] + 1
             Rec[i,j] \leftarrow ``L"
        \mathbf{end}
         else
             D[i,j] \leftarrow D[i-1,j] + 1
             Rec[i,j] \leftarrow ``U"
         end
    \mathbf{end}
\mathbf{end}
```



```
//动态规划
for i \leftarrow 1 to n do
    for j \leftarrow 1 to m do
        c \leftarrow 0
        if s_i \neq t_j then c \leftarrow 1
         end
        replace \leftarrow D[i-1, j-1] + c
         delete \leftarrow D[i-1,j]+1
        insert \leftarrow D[i, j-1] + 1
        if replace = min\{delete, insert, replace\} then
             D[i,j] \leftarrow D[i-1,j-1] + c
             Rec[i,j] \leftarrow "LU"
         end
         else if insert = min\{delete, insert, replace\} then
             D[i,j] \leftarrow D[i,j-1] + 1
             Rec[i,j] \leftarrow "L"
         end
       f else
                                                                                   采取删除操作
            D[i,j] \leftarrow D[i-1,j] + 1
Rec[i,j] \leftarrow ``U"
        end
    end
end
```



```
输入: 矩阵Rec, 字符串s,t, 位置索引i和j
输出: 操作序列
if i = 0 and j = 0 then
\perp return NULL
end
 \begin{array}{c} \textbf{if} \ Rec[i,j] = \text{``}LU\text{''} \ \textbf{then} \\ | \ \operatorname{Print-MED}(Rec,s,t,i-1,j-1) \end{array} 
                                                    采取替换操作
    if s_i = t_i then
     print "无需操作"
    end
    else
     | print "用t_j替换s_i"
    end
end
else if Rec[i, j] = "U" then
    Print-MED(Rec, s, t, i - 1, j)
    print "删除s_i"
end
else
    Print-MED(Rec, s, t, i, j - 1)
    print "插入t_i"
end
```



```
输入: 矩阵Rec, 字符串s,t, 位置索引i和j
输出: 操作序列
if i = 0 and j = 0 then
\perp return NULL
end
if Rec[i, j] = "LU" then
 Print-MED(Rec, s, t, i - 1, j - 1)
                                       递归输出子问题方案
  if s_i = t_j then
   print "无需操作"
  end
   else
    | print "用t_j替换s_i"
  end
end
else if Rec[i, j] = "U" then
  Print-MED(Rec, s, t, i - 1, j)
  print "删除s_i"
end
else
  Print-MED(Rec, s, t, i, j - 1)
  print "插入t_i"
end
```



```
输入: 矩阵Rec, 字符串s,t, 位置索引i和j
输出: 操作序列
if i = 0 and j = 0 then
\perp return NULL
end
if Rec[i,j] = "LU" then
   Print-MED(Rec, s, t, i-1, j-1)
  f if s_i = t_j then print "无需操作"
                                 替换/无操作
 end
 else
 ■ print "用t_j替换s_i"
  end
end
else if Rec[i, j] = "U" then
  Print-MED(Rec, s, t, i - 1, j)
  print "删除s_i"
end
else
  Print-MED(Rec, s, t, i, j - 1)
   print "插入t_i"
end
```



```
输入: 矩阵Rec, 字符串s,t, 位置索引i和j
 输出: 操作序列
if i = 0 and j = 0 then
 | return NULL
 end
if Rec[i, j] = "LU" then
   Print-MED(Rec, s, t, i - 1, j - 1)
   if s_i = t_i then
    │ print "无需操作"
   end
    else
     | print "用t_j替换s_i"
    end
(else if Rec[i,j] = "U" then
                                       采取删除操作
   Print-MED(Rec, s, t, i - 1, j)
   print "删除s_i"
end
else
   Print-MED(Rec, s, t, i, j - 1)
   print "插入t_i"
 end
```



```
输入: 矩阵Rec, 字符串s,t, 位置索引i和j
输出: 操作序列
if i = 0 and j = 0 then
\perp return NULL
end
if Rec[i, j] = "LU" then
   Print-MED(Rec, s, t, i - 1, j - 1)
   if s_i = t_i then
    print "无需操作"
   end
   else
    | print "用t_j替换s_i"
   end
end
else if Rec[i, j] = "U" then
   Print-MED(Rec, s, t, i - 1, j)
   print "删除s_i"
\mathbf{end}
/else
                                        采取插入操作
   Print-MED(Rec, s, t, i, j - 1)
   print "插入t_i"
```

### 时间复杂度分析



```
//动态规划
for i \leftarrow 1 to n do
   for j \leftarrow 1 to m do
       c \leftarrow 0
       if s_i \neq t_j then
        c \leftarrow 1
       end
       replace \leftarrow D[i-1, j-1] + c
       delete \leftarrow D[i-1,j]+1
       insert \leftarrow D[i, j-1] + 1
       if replace = min\{delete, insert, replace\} then
           D[i,j] \leftarrow D[i-1,j-1] + c
                                                                                          -0(m)
           Rec[i,j] \leftarrow "LU"
       end
       else if insert = min\{delete, insert, replace\} then
           D[i,j] \leftarrow D[i,j-1] + 1
           Rec[i,j] \leftarrow "L"
       end
       else
           D[i,j] \leftarrow D[i-1,j] + 1
           Rec[i,j] \leftarrow "U"
       \mathbf{end}
   \mathbf{end}
                                   时间复杂度:O(mn)
end
```

## 小结



#### 最长公共子序列

如果 $s_i \neq t_j$ 

如果 $s_i = t_i$ 

$$C[i,j] = \begin{cases} \max\{C[i-1,j], C[i,j-1]\}, x_i \neq y_j \\ C[i-1,j-1] + 1, x_i = y_j \end{cases}$$

#### 最小编辑距离

删除

$$t \mid B \mid D \mid C \mid A \mid B \mid A$$

$$t \mid B \mid D \mid C \mid A \mid B \mid A$$

$$t \mid B \mid D \mid C \mid A \mid B \mid A$$

$$D[i,j] = \min egin{cases} D[i-1,j] + 1 \ D[i,j-1] + 1 \ D[i-1,j-1] + egin{cases} 0, if \ s[i] = t[j] \ 1, if \ s[i] 
eq t[j] \end{cases}$$





