

1. 采用动态规划技术求解RNA序列: AUGAUGGCCAU
的最大碱基对数目。

A U G A U G G C C A U
1 2 3 4 5 6 7 8 9 10 11

RNA-Secondary-Structure (n, b_1, b_2, \dots, b_n)

For $k = 5$ To $n - 1$

For $i = 1$ To $n - k$

$j \leftarrow i + k.$

For each b_t ($i \leq t < j - 4$) paired with b_j

$T = 1 + M[i, t - 1] + M[t + 1, j - 1].$

$M[i, j] \leftarrow \max\{M[i, j - 1], T\}.$

Return $M[1, n].$

5						
4						
3						
2						
1						
	6	7	8	9	10	11

RNA-Secondary-Structure $(n, b_1, b_2, \dots, b_n)$

```
For  $k = 5$  To  $n - 1$ 
  For  $i = 1$  To  $n - k$ 
     $j \leftarrow i + k$ .
    For each  $b_t$  ( $i \leq t < j - 4$ ) paired with  $b_j$ 
       $T = 1 + M[i, t - 1] + M[t + 1, j - 1]$ .
       $M[i, j] \leftarrow \max\{M[i, j - 1], T\}$ .
Return  $M[1, n]$ .
```

A	U	G	A	U	G	G	C	C	A	U
1	2	3	4	5	6	7	8	9	10	11

$$i \leq t < j - 4$$

5	0	0	0	0		
4	0	0	0			
3	0	0				
2	0					
1						
	6	7	8	9	10	11

5	0	0	0	0	1	
4	0	0	0	0		
3	0	0	1			
2	0	0				
1	0					
	6	7	8	9	10	11

RNA-Secondary-Structure $(n, b_1, b_2, \dots, b_n)$

```

For  $k = 5$  To  $n - 1$ 
  For  $i = 1$  To  $n - k$ 
     $j \leftarrow i + k$ .
    For each  $b_t$  ( $i \leq t < j - 4$ ) paired with  $b_j$ 
       $T = 1 + M[i, t - 1] + M[t + 1, j - 1]$ .
       $M[i, j] \leftarrow \max\{M[i, j - 1], T\}$ .
Return  $M[1, n]$ .
  
```

A	U	G	A	U	G	G	C	C	A	U
1	2	3	4	5	6	7	8	9	10	11

$$i \leq t < j - 4$$

5	0	0	0	0	1	1
4	0	0	0	0	1	
3	0	0	1	1		
2	0	0	1			
1	0	0				
	6	7	8	9	10	11

5	0	0	0	0	1	1
4	0	0	0	0	1	2
3	0	0	1	1	1	
2	0	0	1	1		
1	0	0	1			
	6	7	8	9	10	11

RNA-Secondary-Structure $(n, b_1, b_2, \dots, b_n)$

```
For  $k = 5$  To  $n - 1$ 
  For  $i = 1$  To  $n - k$ 
     $j \leftarrow i + k$ .
    For each  $b_t$  ( $i \leq t < j - 4$ ) paired with  $b_j$ 
       $T = 1 + M[i, t - 1] + M[t + 1, j - 1]$ .
       $M[i, j] \leftarrow \max\{M[i, j - 1], T\}$ .
Return  $M[1, n]$ .
```

A	U	G	A	U	G	G	C	C	A	U
1	2	3	4	5	6	7	8	9	10	11

$$i \leq t < j - 4$$

5	0	0	0	0	1	1
4	0	0	0	0	1	2
3	0	0	1	1	1	2
2	0	0	1	1	2	
1	0	0	1	1		
	6	7	8	9	10	11

5	0	0	0	0	1	1
4	0	0	0	0	1	2
3	0	0	1	1	1	2
2	0	0	1	1	2	2
1	0	0	1	1	2	
	6	7	8	9	10	11

RNA-Secondary-Structure $(n, b_1, b_2, \dots, b_n)$

For $k = 5$ To $n - 1$

For $i = 1$ To $n - k$

$j \leftarrow i + k.$

For each b_t ($i \leq t < j - 4$) paired with b_j

$T = 1 + M[i, t - 1] + M[t + 1, j - 1].$

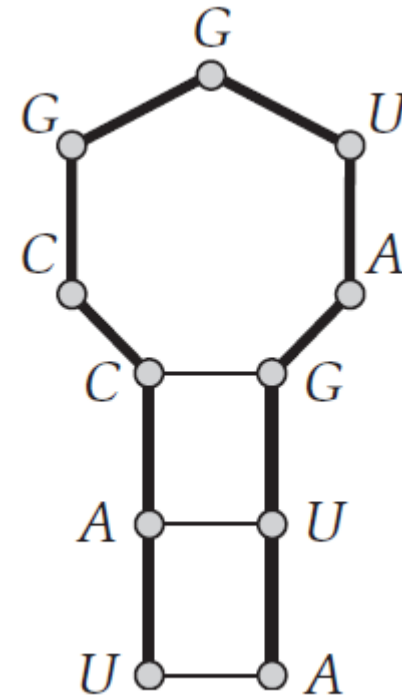
$M[i, j] \leftarrow \max\{M[i, j - 1], T\}.$

Return $M[1, n].$

A	U	G	A	U	G	G	C	C	A	U
1	2	3	4	5	6	7	8	9	10	11

$$i \leq t < j - 4$$

5	0	0	0	0	1	1
4	0	0	0	0	1	2
3	0	0	1	1	1	2
2	0	0	1	1	2	2
1	0	0	1	1	2	3
	6	7	8	9	10	11



1. 给定如下计算A和B的最长公共子序列长度的递推式：

$$L[i, j] = \begin{cases} 0 & \text{若 } i = 0 \text{ 或 } j = 0 \\ L[i - 1, j - 1] + 1 & \text{若 } i > 0, j > 0 \text{ 和 } a_i = b_j \\ \max\{L[i, j - 1], L[i - 1, j]\} & \text{若 } i > 0, j > 0 \text{ 和 } a_i \neq b_j \end{cases}$$

采用动态规划计算实例 **A=“xyxxzxyzxy”** 和
B=“zxzyyzxxxyxxz” 的最长公共子序列。

2. 给定如下计算A和B的最长公共子序列长度的递推式：

$$L[i, j] = \begin{cases} 0 & \text{若 } i = 0 \text{ 或 } j = 0 \\ L[i - 1, j - 1] + 1 & \text{若 } i > 0, j > 0 \text{ 和 } a_i = b_j \\ \max\{L[i, j - 1], L[i - 1, j]\} & \text{若 } i > 0, j > 0 \text{ 和 } a_i \neq b_j \end{cases}$$

采用动态规划技术计算实例A="xyxxzxyzxy"和B="zxzyyzxxxyxxz"的最长公共子序列。

LCS算法

输入：字符串A和B； 输出： A和B最长公共子序列的长度。

For $i \leftarrow 0$ to n

$L[i, 0] \leftarrow 0$

For $j \leftarrow 0$ to m

$L[0, j] \leftarrow 0$

For $i \leftarrow 1$ to n

 For $j \leftarrow 1$ to m

 If $a_i = b_j$ Then $L[i, j] \leftarrow L[i - 1, j - 1] + 1$

 Else $L[i, j] \leftarrow \max\{L[i, j - 1], L[i - 1, j]\}$

Return $L[n, m]$

2. 给定如下计算A和B的最长公共子序列长度的递推式:

$$L[i, j] = \begin{cases} 0 & \text{若 } i = 0 \text{ 或 } j = 0 \\ L[i - 1, j - 1] + 1 & \text{若 } i > 0, j > 0 \text{ 和 } a_i = b_j \\ \max\{L[i, j - 1], L[i - 1, j]\} & \text{若 } i > 0, j > 0 \text{ 和 } a_i \neq b_j \end{cases}$$

采用动态规划技术计算实例A="xyxxzxyzxy"和B="zxzyyzxxxyxxz"的最长公共子序列。

[illegible]

2. 给定如下计算A和B的最长公共子序列长度的递推式:

$$L[i, j] = \begin{cases} 0 & \text{若 } i = 0 \text{ 或 } j = 0 \\ L[i - 1, j - 1] + 1 & \text{若 } i > 0, j > 0 \text{ 和 } a_i = b_j \\ \max\{L[i, j - 1], L[i - 1, j]\} & \text{若 } i > 0, j > 0 \text{ 和 } a_i \neq b_j \end{cases}$$

采用动态规划技术计算实例A="xyxxzxyzxy"和B="zxzyyzxxxyxxz"的最长公共子序列。

[illegible]

2. 给定如下计算A和B的最长公共子序列长度的递推式:

$$L[i, j] = \begin{cases} 0 & \text{若 } i = 0 \text{ 或 } j = 0 \\ L[i - 1, j - 1] + 1 & \text{若 } i > 0, j > 0 \text{ 和 } a_i = b_j \\ \max\{L[i, j - 1], L[i - 1, j]\} & \text{若 } i > 0, j > 0 \text{ 和 } a_i \neq b_j \end{cases}$$

采用动态规划技术计算实例A="xyxxzxyzxy"和B="zxzyyzxxxyxxz"的最长公共子序列。

[illegible]

2. 给定如下计算A和B的最长公共子序列长度的递推式：

$$L[i, j] = \begin{cases} 0 & \text{若 } i = 0 \text{ 或 } j = 0 \\ L[i - 1, j - 1] + 1 & \text{若 } i > 0, j > 0 \text{ 和 } a_i = b_j \\ \max\{L[i, j - 1], L[i - 1, j]\} & \text{若 } i > 0, j > 0 \text{ 和 } a_i \neq b_j \end{cases}$$

采用动态规划技术计算实例A=“xyxxzxyzxy”和B=“zxzyyzxxxyxxz”的最长公共子序列。

	0	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	1	1	1	1	1	1	1	1	1	1
2	0	0	1	1	2	2	2	2	2	2	2	2	2
3	0	0	1	1	2	2	2	3	3	3	3	3	3
4	0	0	1	1	2	2	2	3	4	4	4	4	4
5	0	1	1	2	2	2	3	3	4	4	4	4	5
6	0	1	2	2	2	2	3	4	4	4	5	5	5
7	0	1	2	2	3	3	3	4	4	5	5	5	5
8	0	1	2	3	3	3	4	4	4	5	5	5	6
9	0	1	2	3	3	3	4	5	5	5	6	6	6
10	0	1	2	3	4	4	4	5	5	6	6	6	6