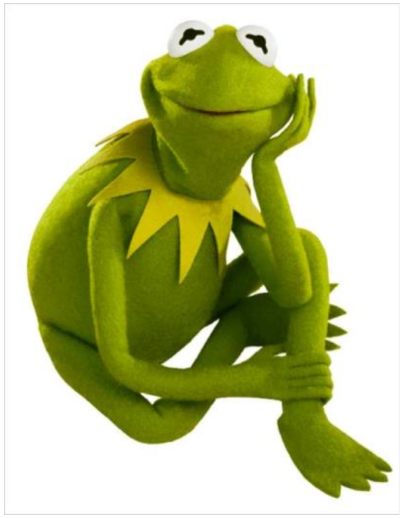
Longest Common Subsequence

Longest Common Subsequence

- How similar are these two species?

DNA:
AGCCCTAAGGGCTACCTAGCTT

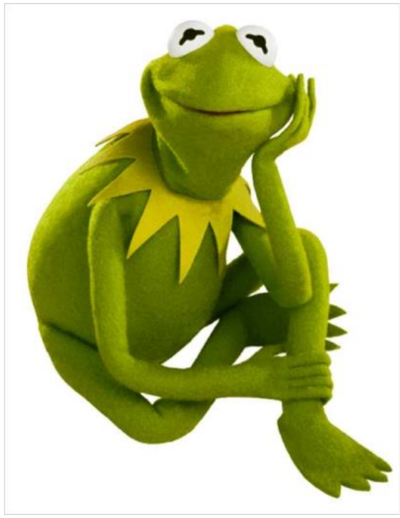


DNA:
GACAGCCTACAAGCGTTAGCTTG

Longest Common Subsequence

- How similar are these two species?

DNA:
AGCCCTAAGGGCTACCTAGCTT



DNA:
GACAGCCTACAAGCGTTAGCTTG

- Pretty similar, their DNA has a long common subsequence:

AGCCTAAGCTTAGCTT

Longest Common Subsequence

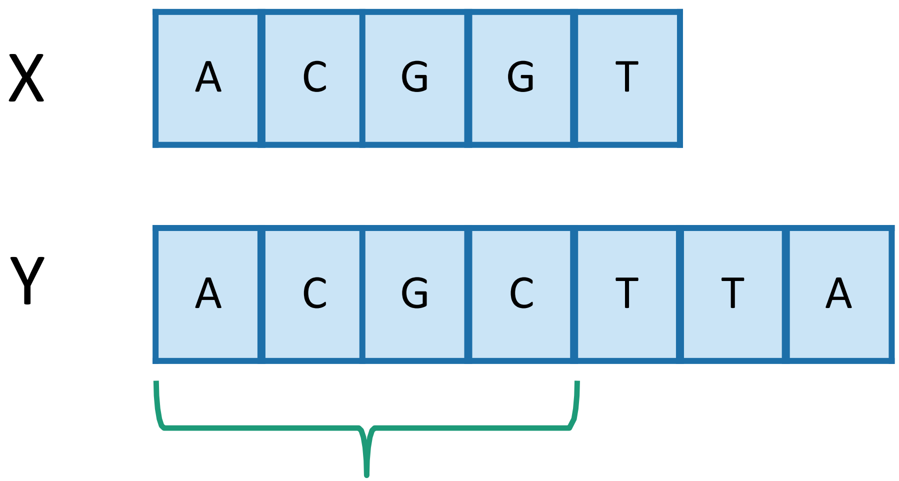
- Subsequence:
  - BDFH is a **subsequence** of ABCDEFGH
- If X and Y are sequences, a **common subsequence**
- is a sequence which is a subsequence of both.
  - BDFH is a **common subsequence** of ABCDEFGH and of ABDFGHI
- A **longest common subsequence**…
  - …is a common subsequence that is longest.
  - The **longest common subsequence** of ABCDEFGH and ABDFGHI is ABDFGH.

Steps for applying Dynamic Programming

- **Step 1:** Identify optimal substructure.
- **Step 2:** Find a recursive formulation for the length of the longest common subsequence.
- **Step 3:** Use dynamic programming to find the length of the longest common subsequence.
- **Step 4:** If needed, keep track of some additional info so that the algorithm from Step 3 can find the actual LCS.

Step 1: Optimal substructure

Prefixes:



Notation: denote this prefix **ACGC** by Y_4

- Our sub-problems will be finding LCS's of prefixes to X and Y.
- Let C[i,j] = length_of_LCS(X_i, Y_j)

Examples: C[2,3] = 2
C[4,4] = 3

Optimal substructure ctd.

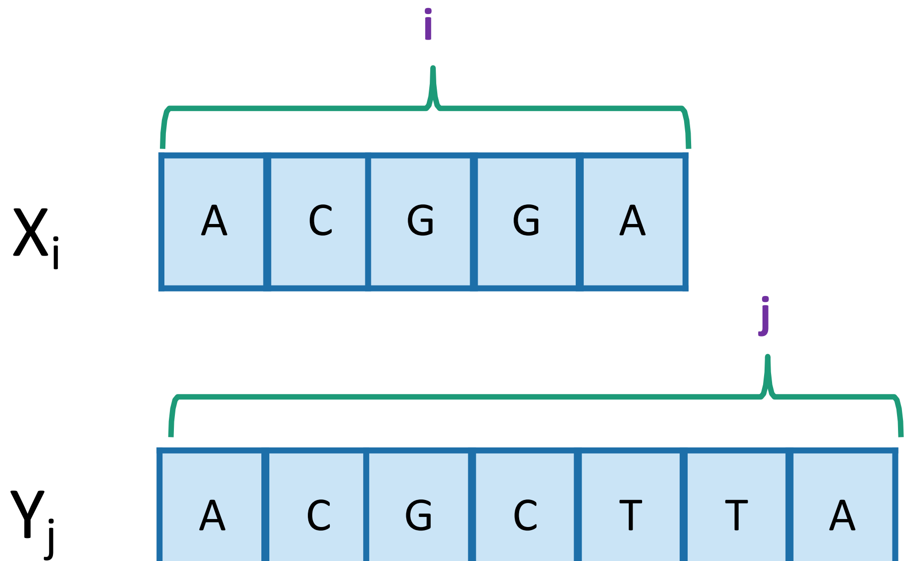
- Subproblem:
  - finding LCS's of prefixes of X and Y.
- Why is this a good choice?
  - As we will see, there's some relationship between LCS's of prefixes and LCS's of the whole things.
  - These subproblems overlap a lot.

Steps for applying Dynamic Programming

- **Step 1:** Identify optimal substructure.
- **Step 2:** Find a recursive formulation for the length of the longest common subsequence.
- **Step 3:** Use dynamic programming to find the length of the longest common subsequence.
- **Step 4:** If needed, keep track of some additional info so that the algorithm from Step 3 can find the actual LCS.

Goal

- Write C[i,j] in terms of the solutions to smaller subproblems

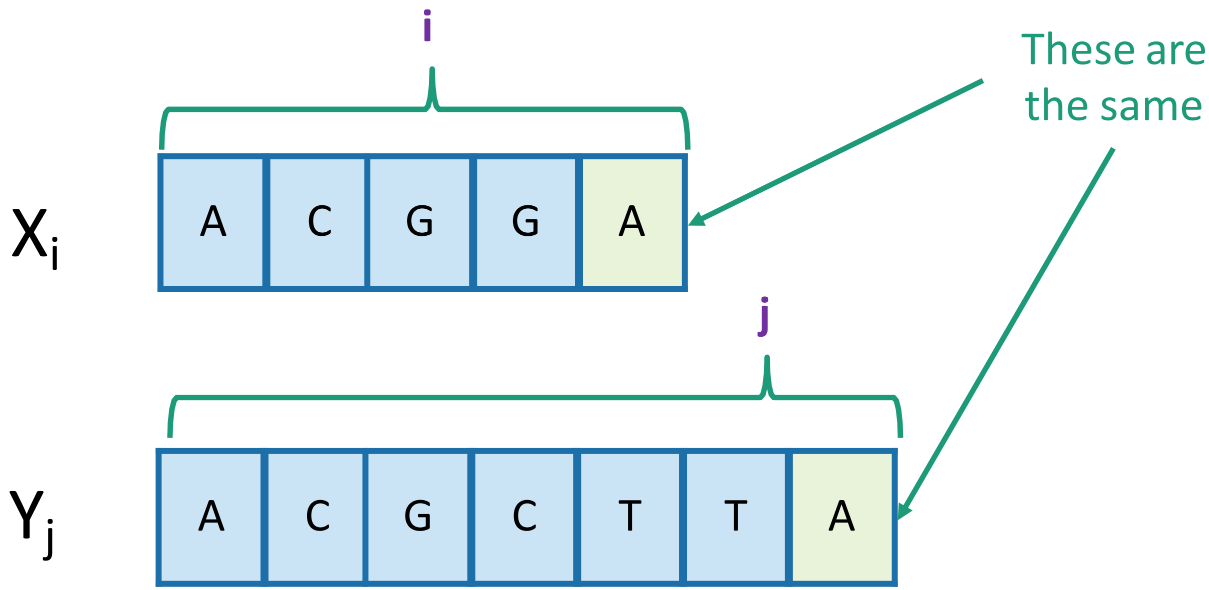


$C[i,j] = \text{length\_of\_LCS}(\ X_i, Y_j\ )$

Two cases

- Our sub-problems will be finding LCS's of prefixes to X and Y.
- Let C[i,j] = length_of_LCS(X_i, Y_j)

Case 1: X[i] = Y[j]

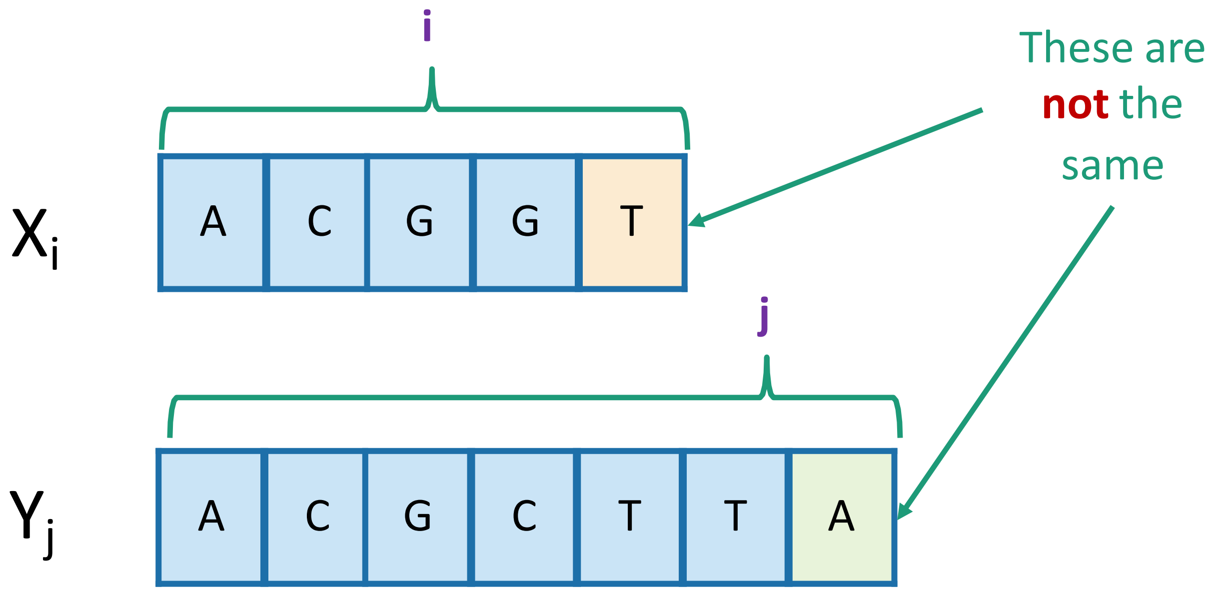


- Then C[i,j] = 1 + C[i-1,j-1].
  - because LCS(X_i,Y_j) = LCS(X_{i-1},Y_{j-1}) followed by $\boxed{A}$

Two cases

- Our sub-problems will be finding LCS's of prefixes to X and Y.
- Let C[i,j] = length_of_LCS(X_i, Y_j)

Case 2: X[i] != Y[j]



- Then C[i,j] = max{ C[i-1,j], C[i,j-1] }.
  - either LCS(X_i,Y_j) = LCS(X_{i-1},Y_j) and $\boxed{T}$ is not involved,
  - or LCS(X_i,Y_j) = LCS(X_i,Y_{j-1}) and $\boxed{A}$ is not involved,
  - (maybe both are not involved, that's covered by the "or")

Recursive formulation of the optimal solution

X_0

Y_j	A	C	G	C	T	T	A

Case 0

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i, j > 0 \\ \max\{\, C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i, j > 0 \end{cases}$$

Case 1

X_i	A	C	G	G	A		
Y_j	A	C	G	C	T	T	A

Case 2

X_i	A	C	G	G	T		
Y_j	A	C	G	C	T	T	A

Steps for applying Dynamic Programming

- **Step 1:** Identify optimal substructure.
- **Step 2:** Find a recursive formulation for the length of the longest common subsequence.
- **Step 3:** Use dynamic programming to find the length of the longest common subsequence.
- **Step 4:** If needed, keep track of some additional info so that the algorithm from Step 3 can find the actual LCS.

LCS DP

- **LCS**(X, Y):
  - C[i,0] = C[0,j] = 0 for all i = 0,…,m, j=0,…n.
  - **For** i = 1,…,m and j = 1,…,n:
    - **If** X[i] = Y[j]:
      - C[i,j] = C[i-1,j-1] + 1
      - B[i,j] = "↖"
    - **Else:**
      - If C[i-1, j] >= C[i, j-1]
        - C[i, j] = C[i-1, j]
        - B[i, j] = "↑"
      - Else:
        - C[i,j] = C[i, j-1]
        - B[i,j] = "←"

  - Return C and B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{\ C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

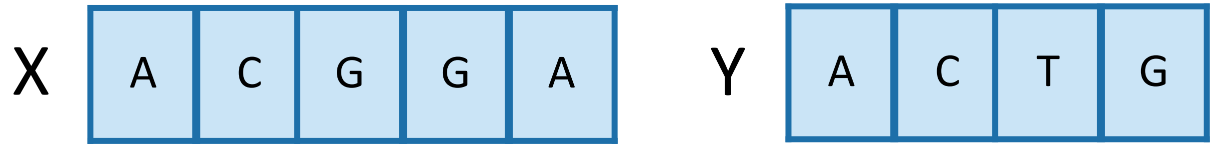
LCS DP

- **LCS**(X, Y):
  - C[i,0] = C[0,j] = 0 for all i = 0,…,m, j=0,…n.
  - **For** i = 1,…,m and j = 1,…,n:
    - **If** X[i] = Y[j]:
      - C[i,j] = C[i-1,j-1] + 1
      - B[i,j] = "↖"
    - **Else:**
      - If C[i-1, j] >= C[i, j-1]
        - C[i, j] = C[i-1, j]
        - B[i, j] = "↑"
      - Else:
        - C[i,j] = C[i, j-1]
        - B[i,j] = "←"
  - Return C and B

Running time: O(nm)

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i, j > 0 \\ \max\{\, C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i, j > 0 \end{cases}$$

Example



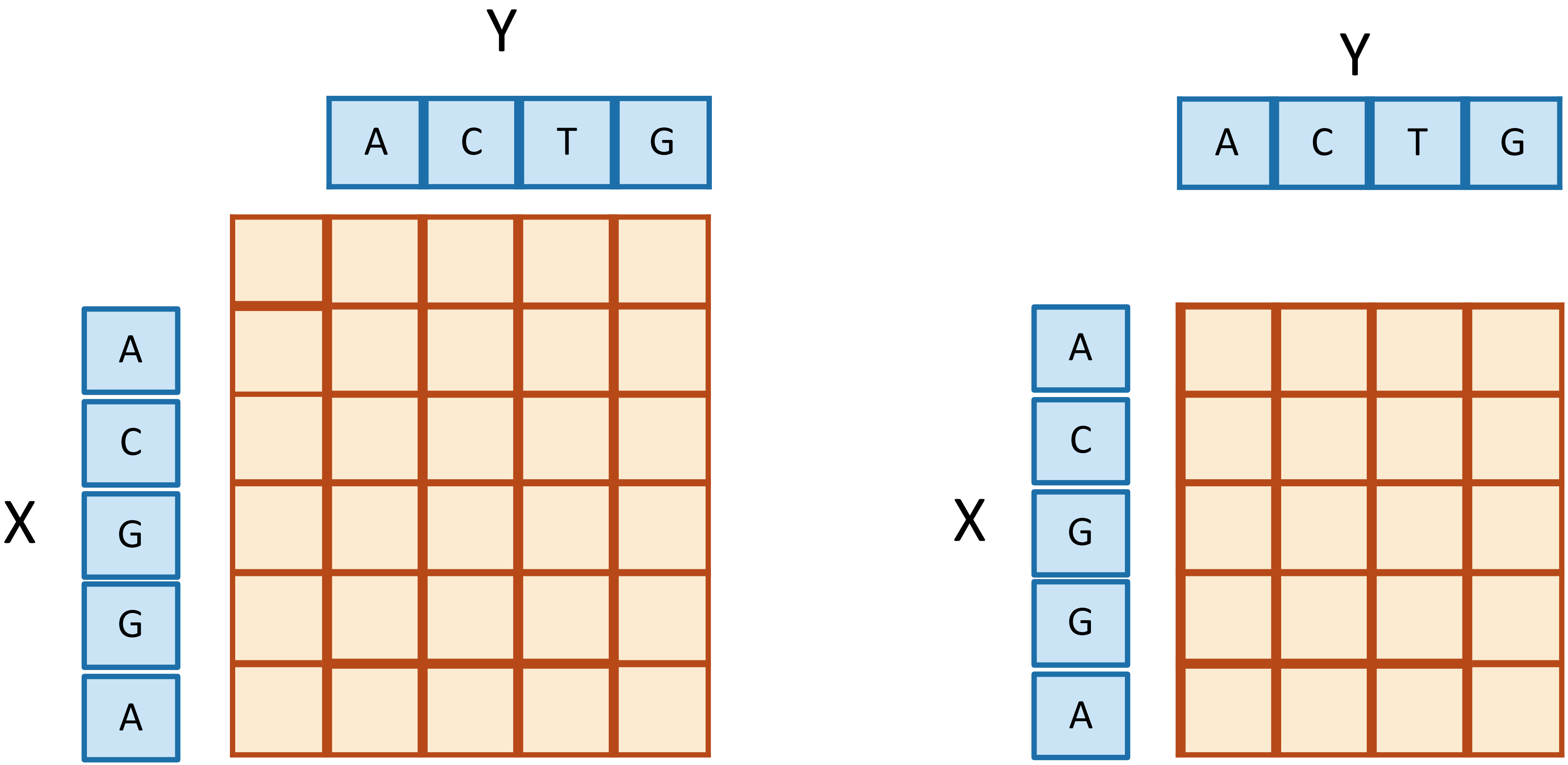


Table C **Table B**

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

		Y: A	C	T	G
	0	0	0	0	0
X: A	0				
C	0				
G	0				
G	0				
A	0				

Table C

	Y: A	C	T	G
X: A				
C				
G				
G				
A				

Table B

$$C[i,j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ C[i-1, j-1] + 1 & \text{if } X[i] = Y[j] \text{ and } i, j > 0 \\ \max\{ C[i, j-1], C[i-1, j] \} & \text{if } X[i] \neq Y[j] \text{ and } i, j > 0 \end{cases}$$

Example

X	A	C	G	G	A

Y	A	C	T	G

		Y: A	C	T	G
X	0	0	0	0	0
A	0				
C	0				
G	0				
G	0				
A	0				

Table C

X \ Y	A	C	T	G
A				
C				
G				
G				
A				

Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{\, C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

Table C

X \ Y		A	C	T	G
	0	0	0	0	0
A	0	1			
C	0				
G	0				
G	0				
A	0				

Table B

X \ Y	A	C	T	G
A	↖			
C				
G				
G				
A				

$$C[i,j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ C[i-1, j-1] + 1 & \text{if } X[i] = Y[j] \text{ and } i,j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & \text{if } X[i] \neq Y[j] \text{ and } i,j > 0 \end{cases}$$

Example

X: A C G G A

Y: A C T G

Y

X		A	C	T	G
	0	0	0	0	0
A	0	1			
C	0				
G	0				
G	0				
A	0				

Table C

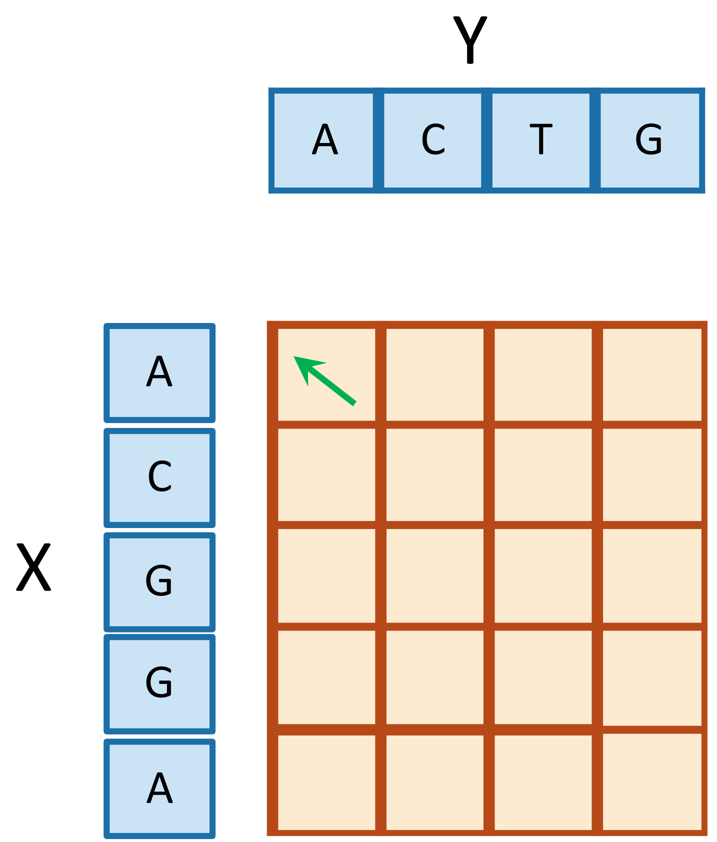


Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{\, C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

X \ Y		A	C	T	G
	0	0	**0**	0	0
A	0	**1**	1		
C	0				
G	0				
G	0				
A	0				

Table C

X \ Y	A	C	T	G
A	↖			
C				
G				
G				
A				

Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{\, C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

X \ Y		A	C	T	G
	0	0	0	0	0
A	0	1	1		
C	0				
G	0				
G	0				
A	0				

Table C

X \ Y	A	C	T	G
A	↖	←		
C				
G				
G				
A				

Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{ C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

X \ Y		A	C	T	G
	0	0	0	0	0
A	0	1	1		
C	0				
G	0				
G	0				
A	0				

Table C

X \ Y	A	C	T	G
A	↖	←		
C				
G				
G				
A				

Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{\ C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

X \ Y		A	C	T	G
	0	0	0	0	0
A	0	1	1	1	
C	0				
G	0				
G	0				
A	0				

Table C

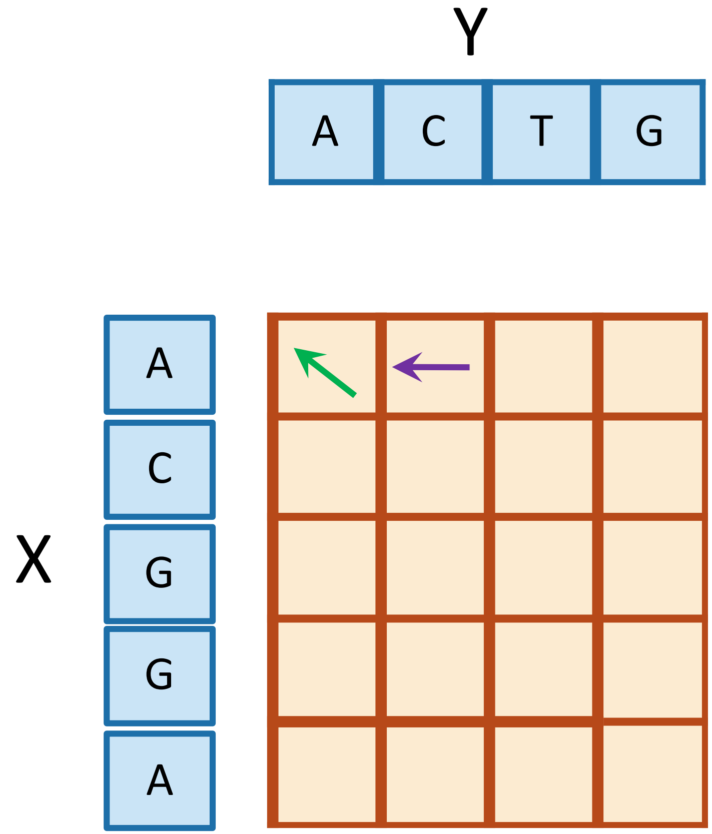


Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X	A	C	G	G	A

Y	A	C	T	G

		Y: A	C	T	G
	0	0	0	0	0
X: A	0	1	1	1	
C	0				
G	0				
G	0				
A	0				

Table C

	Y: A	C	T	G
X: A	↖	←	←	
C				
G				
G				
A				

Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{\ C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X	A	C	G	G	A

Y	A	C	T	G

Table C

X \ Y		A	C	T	G
	0	0	0	0	**0**
A	0	1	1	**1**	
C	0				
G	0				
G	0				
A	0				

Table B

X \ Y	A	C	T	G
A	↖	←	←	
C				
G				
G				
A				

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{\ C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

X \ Y		A	C	T	G
	0	0	0	0	**0**
A	0	1	1	**1**	1
C	0				
G	0				
G	0				
A	0				

Table C

X \ Y	A	C	T	G
A	↖	←	←	←
C				
G				
G				
A				

Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: A C G G A

Y: A C T G

Table C

X \ Y		A	C	T	G
	0	0	0	0	0
A	0	**1**	1	**1**	1
C	**0**				
G	0				
G	0				
A	0				

Table B

X \ Y	A	C	T	G
A	↖	←	←	←
C				
G				
G				
A				

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1,j-1]+1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{C[i,j-1], C[i-1,j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

Table C

		Y: A	C	T	G
	0	0	0	0	0
X: A	0	**1**	1	**1**	1
C	**0**	1			
G	0				
G	0				
A	0				

Table B

	Y: A	C	T	G
X: A	↖	←	←	←
C	↑			
G				
G				
A				

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{\ C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X	A	C	G	G	A

Y	A	C	T	G

Y

		A	C	T	G
	0	0	0	0	0
A	0	1	**1**	**1**	1
C	0	**1**			
G	0				
G	0				
A	0				

X

Table C

Y

	A	C	T	G
A	↖	←	←	←
C	↑			
G				
G				
A				

X

Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1,j-1]+1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{C[i,j-1], C[i-1,j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X	A	C	G	G	A

Y	A	C	T	G

Table C

X \ Y		A	C	T	G
	0	0	0	0	0
A	0	**1**	1	**1**	1
C	0	1	2		
G	0				
G	0				
A	0				

Table B

X \ Y	A	C	T	G
A	↖	←	←	←
C	↑	↖		
G				
G				
A				

$$C[i,j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ C[i-1, j-1] + 1 & \text{if } X[i] = Y[j] \text{ and } i,j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & \text{if } X[i] \neq Y[j] \text{ and } i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

Table C

		Y: A	C	T	G
	0	0	0	0	0
X: A	0	1	1	**1**	1
C	0	1	**2**		
G	0				
G	0				
A	0				

Table B

	Y: A	C	T	G
X: A	↖	←	←	←
C	↑	↖		
G				
G				
A				

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{\ C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

Y

X		A	C	T	G
	0	0	0	0	0
A	0	1	1	**1**	1
C	0	1	**2**	2	
G	0				
G	0				
A	0				

Table C

Y

X	A	C	T	G
A	↖	←	←	←
C	↑	↖		
G				
G				
A				

Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{\, C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

		Y: A	C	T	G
X	0	0	0	0	0
A	0	1	1	**1**	1
C	0	1	**2**	2	
G	0				
G	0				
A	0				

Table C

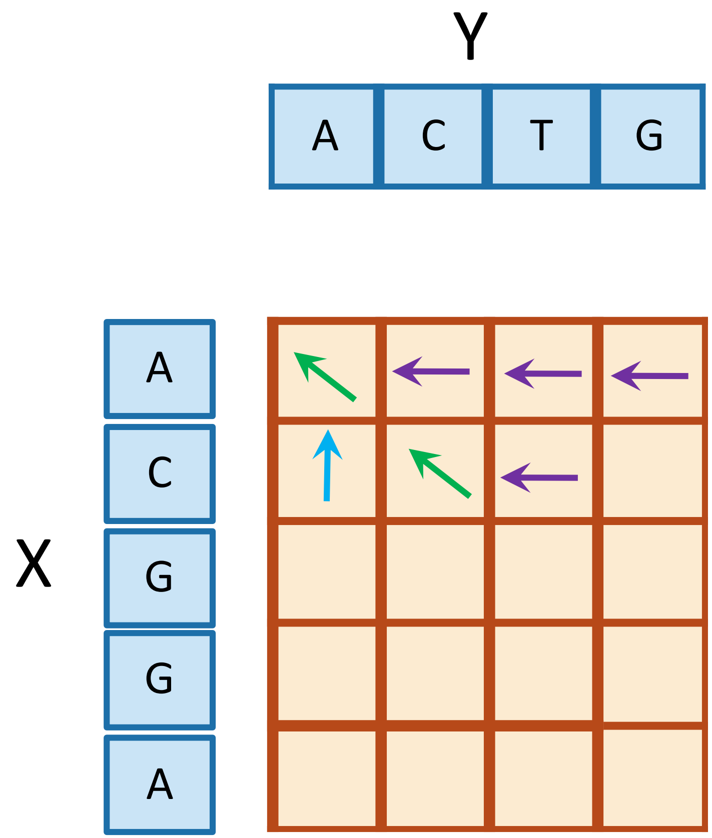


Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

X \ Y		A	C	T	G
	0	0	0	0	0
A	0	1	1	1	**1**
C	0	1	2	**2**	
G	0				
G	0				
A	0				

Table C

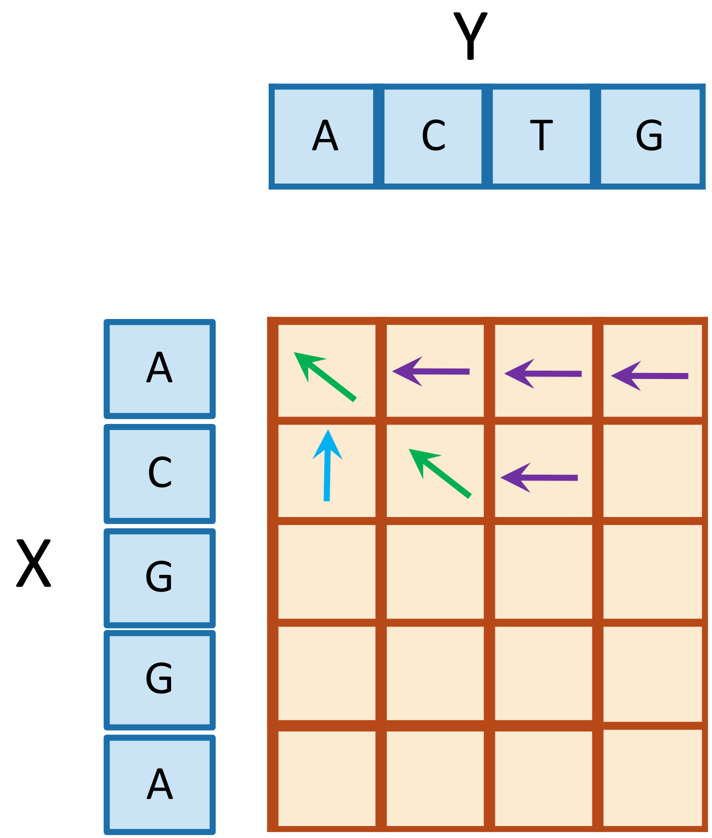


Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{\ C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

		A	C	T	G
	0	0	0	0	0
A	0	1	1	1	**1**
C	0	1	2	**2**	2
G	0				
G	0				
A	0				

Table C

	A	C	T	G
A	↖	←	←	←
C	↑	↖	←	
G				
G				
A				

Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X	A	C	G	G	A

Y	A	C	T	G

		Y: A	C	T	G
	0	0	0	0	0
X: A	0	1	1	1	**1**
C	0	1	2	**2**	2
G	0				
G	0				
A	0				

Table C

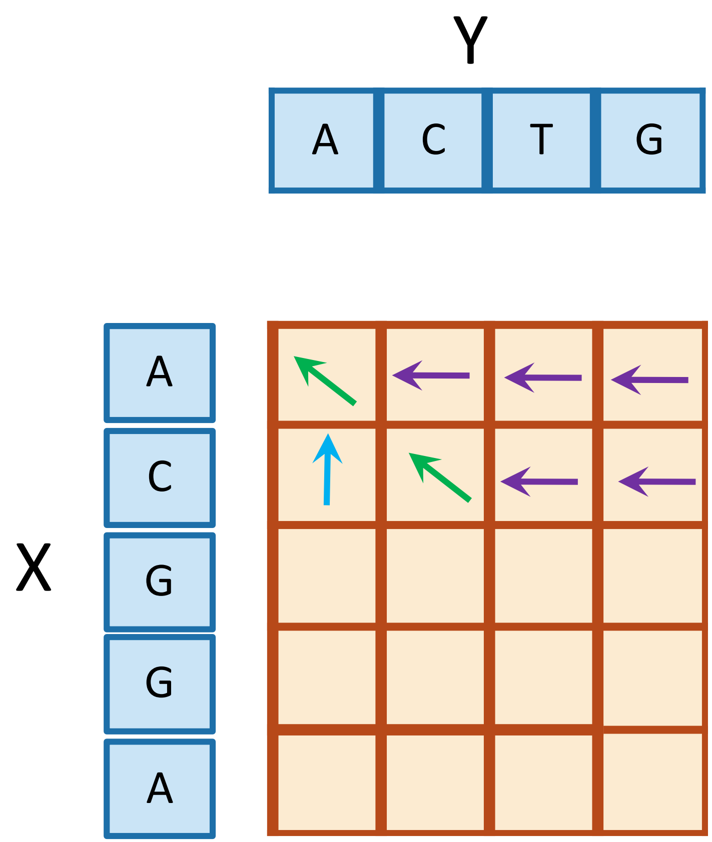


Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X

A	C	G	G	A

Y

A	C	T	G

		Y: A	C	T	G
	0	0	0	0	0
X: A	0	1	1	1	1
C	0	**1**	2	2	2
G	**0**	1			
G	0				
A	0				

Table C

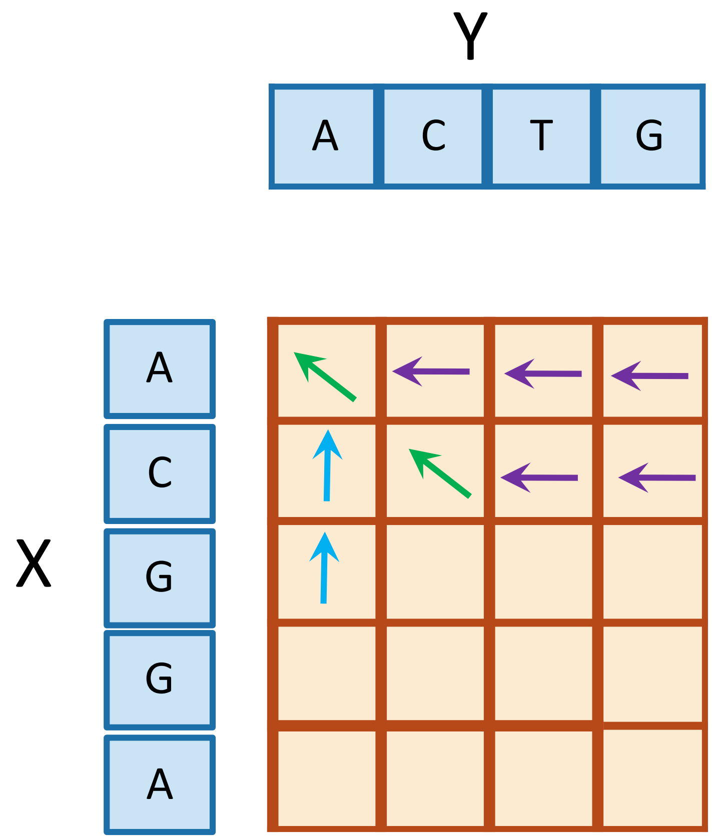


Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{\, C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: A C G G A

Y: A C T G

		Y: A	C	T	G
	0	0	0	0	0
X: A	0	1	1	1	1
C	0	1	**2**	2	2
G	0	**1**	2		
G	0				
A	0				

Table C

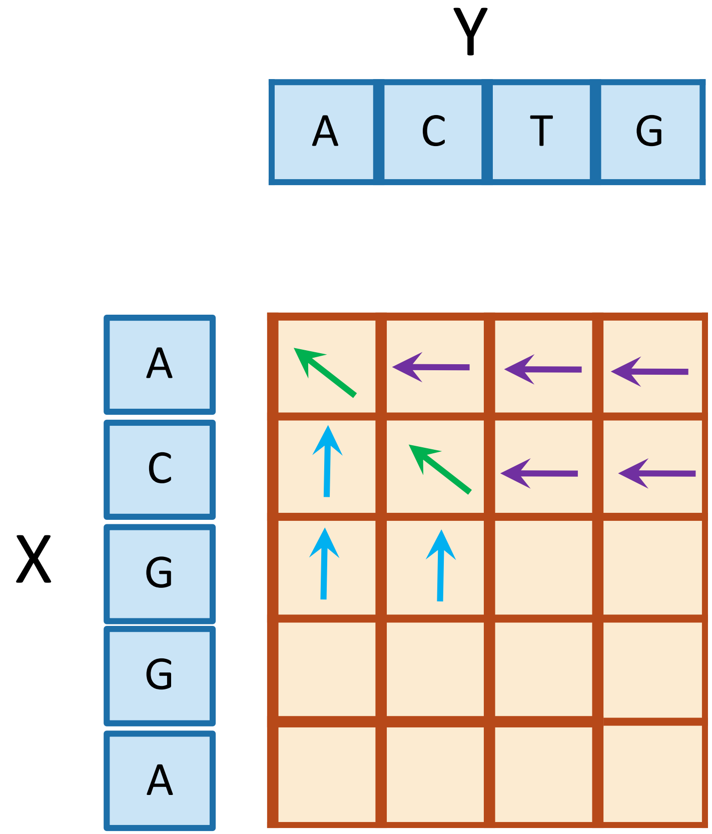


Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{C[i,j-1], C[i-1,j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: A C G G A

Y: A C T G

X \ Y		A	C	T	G
	0	0	0	0	0
A	0	1	1	1	1
C	0	1	2	**2**	2
G	0	1	**2**	2	
G	0				
A	0				

Table C

X \ Y	A	C	T	G
A	↖	←	←	←
C	↑	↖	←	←
G	↑	↑	↑	
G				
A				

Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: A C G G A

Y: A C T G

Table C

		Y: A	C	T	G
	0	0	0	0	0
X: A	0	1	1	1	1
C	0	1	2	2	**2**
G	0	1	2	**2**	2
G	0				
A	0				

Table B

	Y: A	C	T	G
X: A	↖	←	←	←
C	↑	↖	←	←
G	↑	↑	↑	↑
G				
A				

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

		A	C	T	G
	0	0	0	0	0
A	0	1	1	1	1
C	0	1	2	2	2
G	0	1	2	2	2
G	0	1	2	2	
A	0				

Table C

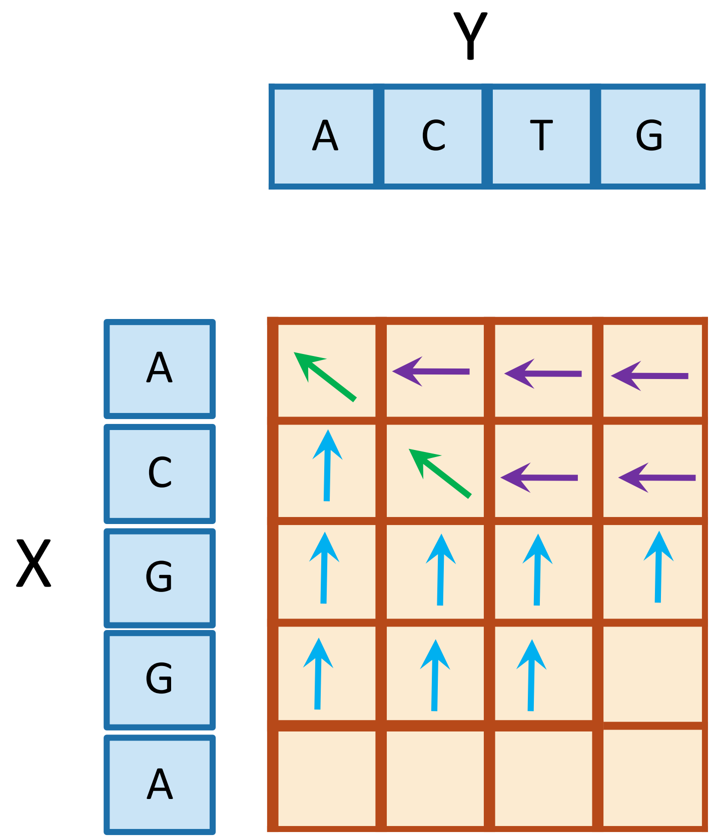


Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: A C G G A

Y: A C T G

Table C

		Y: A	C	T	G
X	0	0	0	0	0
A	0	1	1	1	1
C	0	1	2	2	2
G	0	1	2	**2**	2
G	0	1	2	2	3
A	0				

Table B

X \ Y	A	C	T	G
A	↖	←	←	←
C	↑	↖	←	←
G	↑	↑	↑	↑
G	↑	↑	↑	↖
A				

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i,j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & if\ X[i] \neq Y[j] and\ i,j > 0 \end{cases}$$

Example

X: A C G G A

Y: A C T G

		Y: A	C	T	G
	0	0	0	0	0
X: A	0	1	1	1	1
C	0	1	2	2	2
G	0	1	2	2	2
G	**0**	1	2	2	3
A	0	1			

Table C

	Y: A	C	T	G
X: A	↖	←	←	←
C	↑	↖	←	←
G	↑	↑	↑	↑
G	↑	↑	↑	↖
A	↖			

Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i, j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & if X[i] \neq Y[j] and\ i, j > 0 \end{cases}$$

Example

X: | A | C | G | G | A |

Y: | A | C | T | G |

Table C

		Y: A	C	T	G
X	0	0	0	0	0
A	0	1	1	1	1
C	0	1	2	2	2
G	0	1	2	2	2
G	0	1	2	2	3
A	0	1	2	2	3

Table B

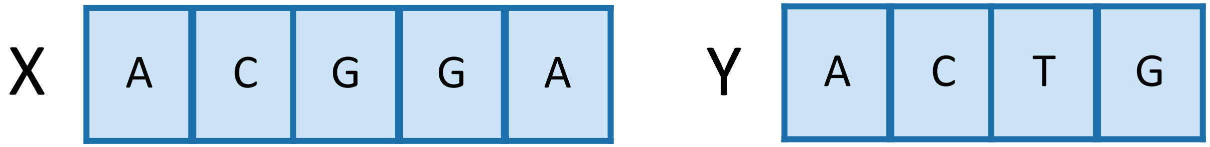
X \ Y	A	C	T	G
A	↖	←	←	←
C	↑	↖	←	←
G	↑	↑	↑	↑
G	↑	↑	↑	↖
A	↖	↑	↑	↑

$$C[i,j] = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ C[i-1, j-1] + 1 & \text{if } X[i] = Y[j] \text{ and } i,j > 0 \\ \max\{C[i, j-1], C[i-1, j]\} & \text{if } X[i] \neq Y[j] \text{ and } i,j > 0 \end{cases}$$

Steps for applying Dynamic Programming

- **Step 1:** Identify optimal substructure.
- **Step 2:** Find a recursive formulation for the length of the longest common subsequence.
- **Step 3:** Use dynamic programming to find the length of the longest common subsequence.
- **Step 4:** If needed, keep track of some additional info so that the algorithm from Step 3 can find the actual LCS.

Example



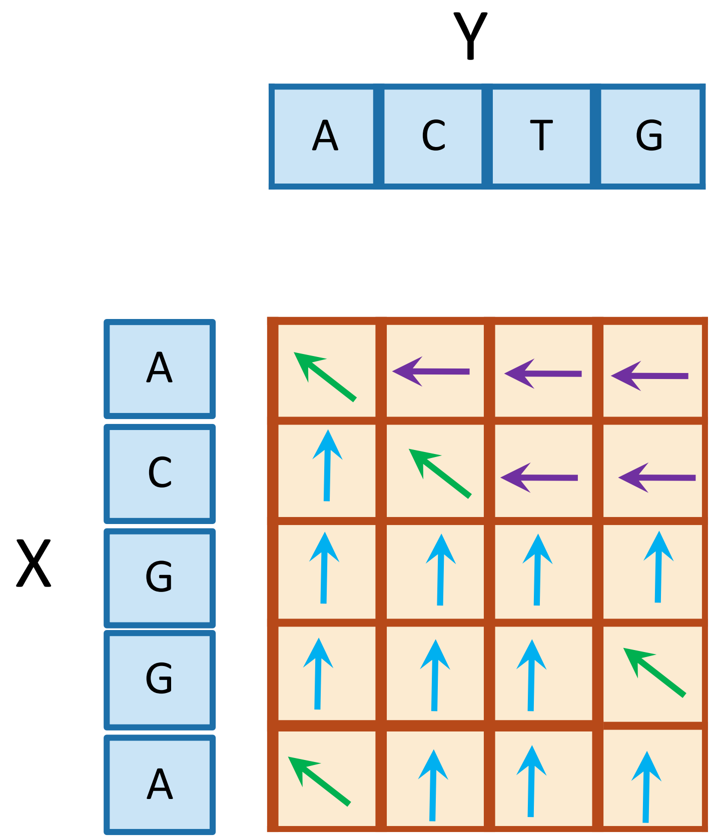


Table B

Example

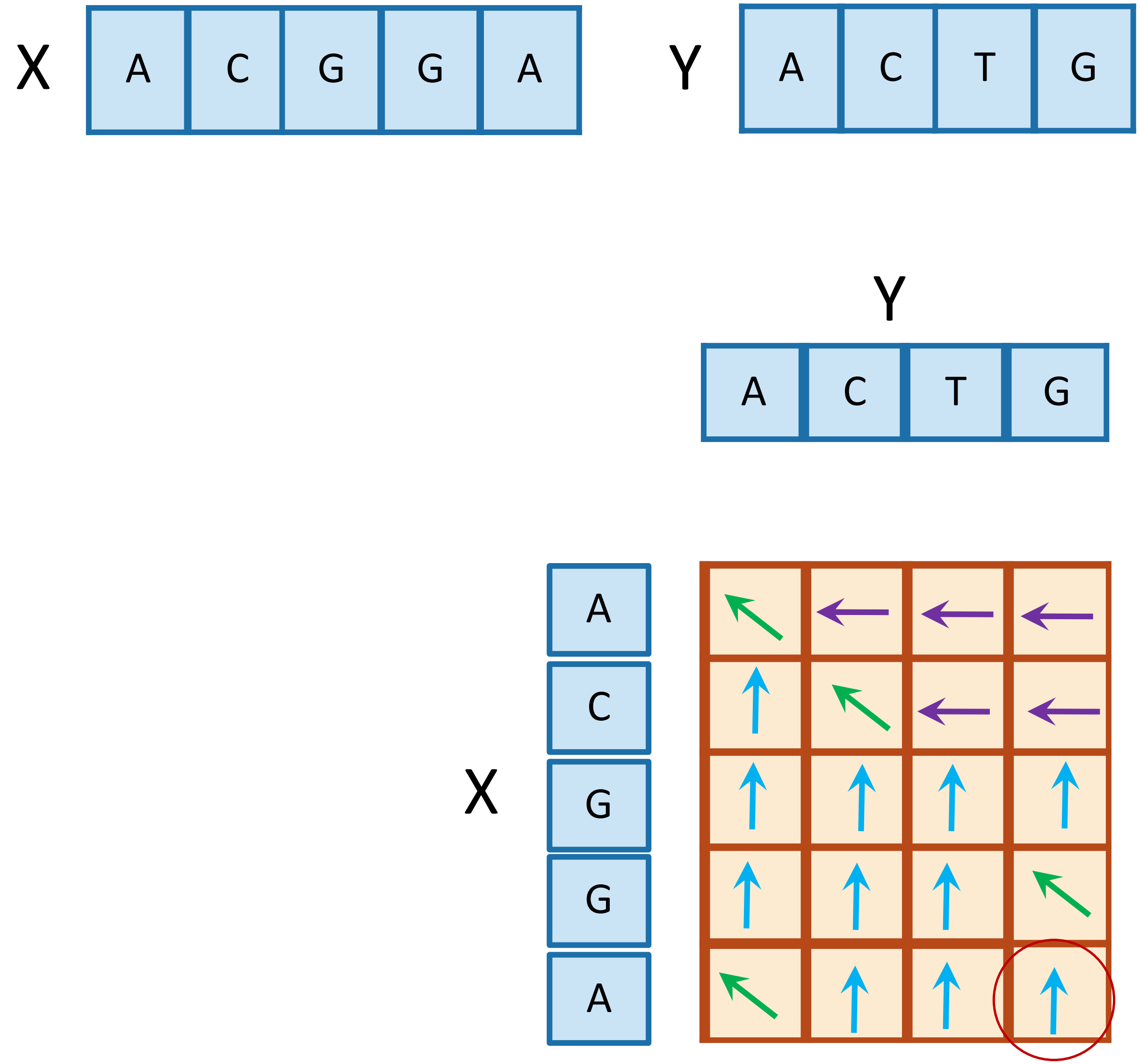


Table B

Example

X	A	C	G	G	A

Y	A	C	T	G

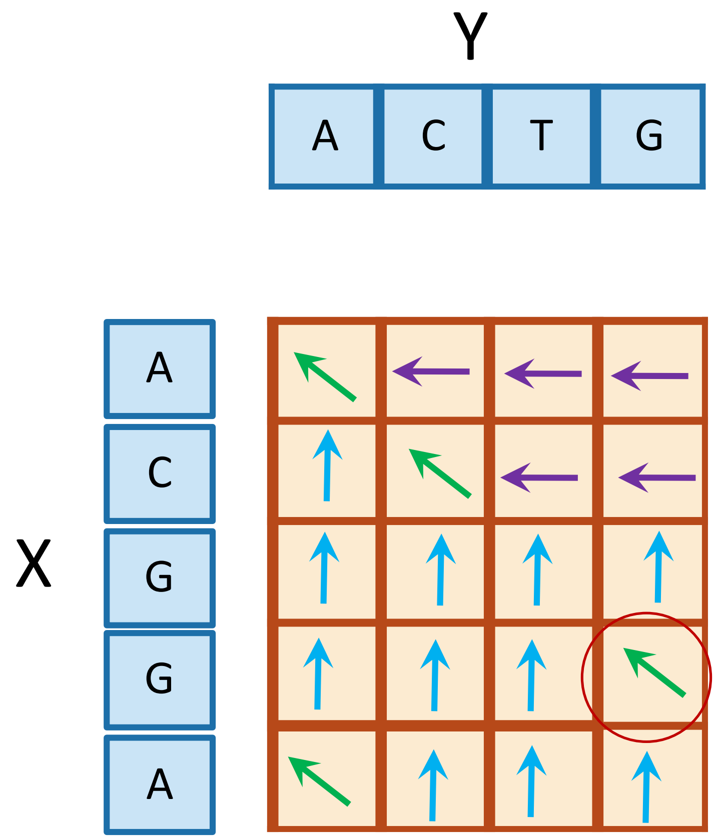


Table B

Example

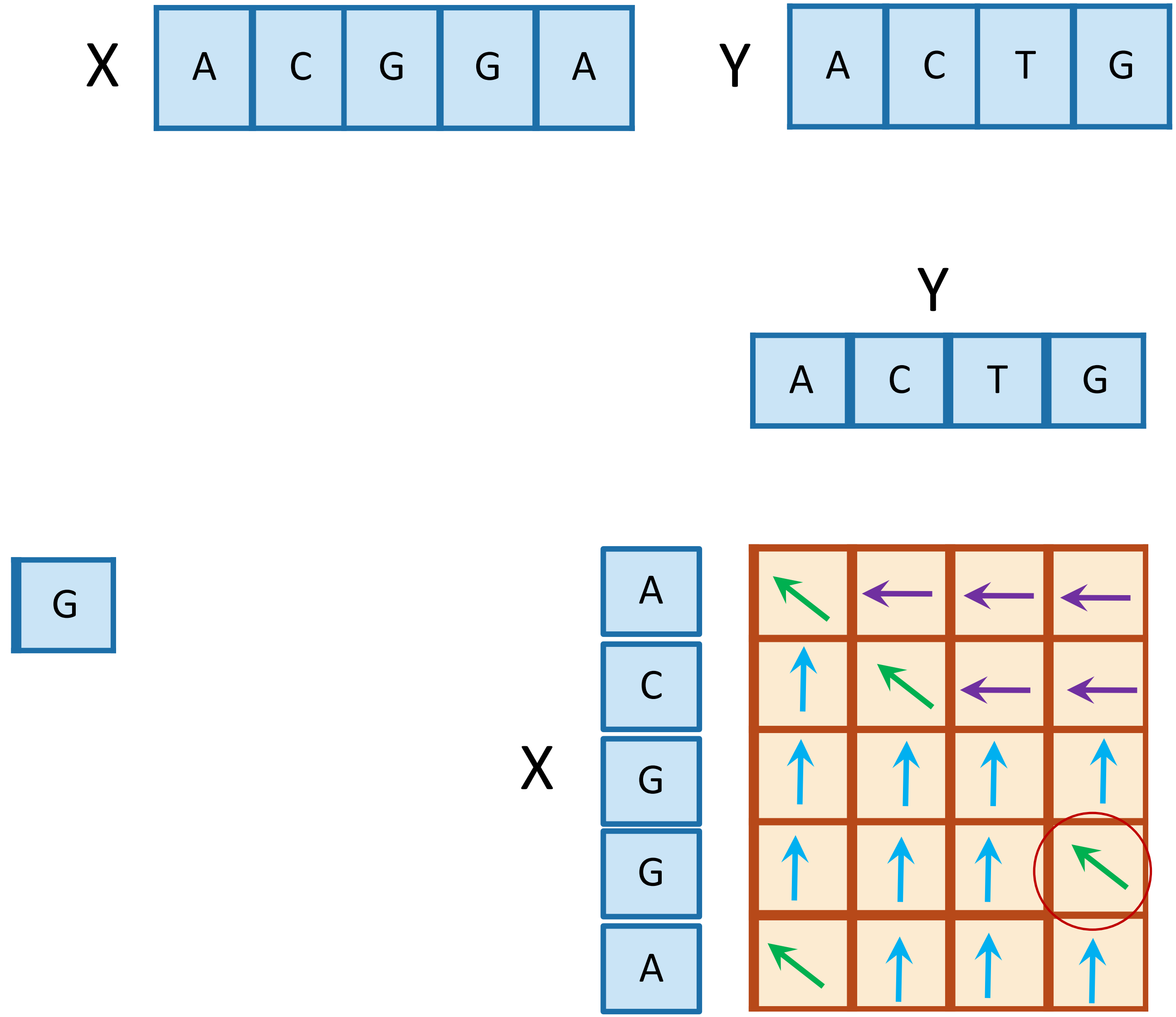


Table B

Example

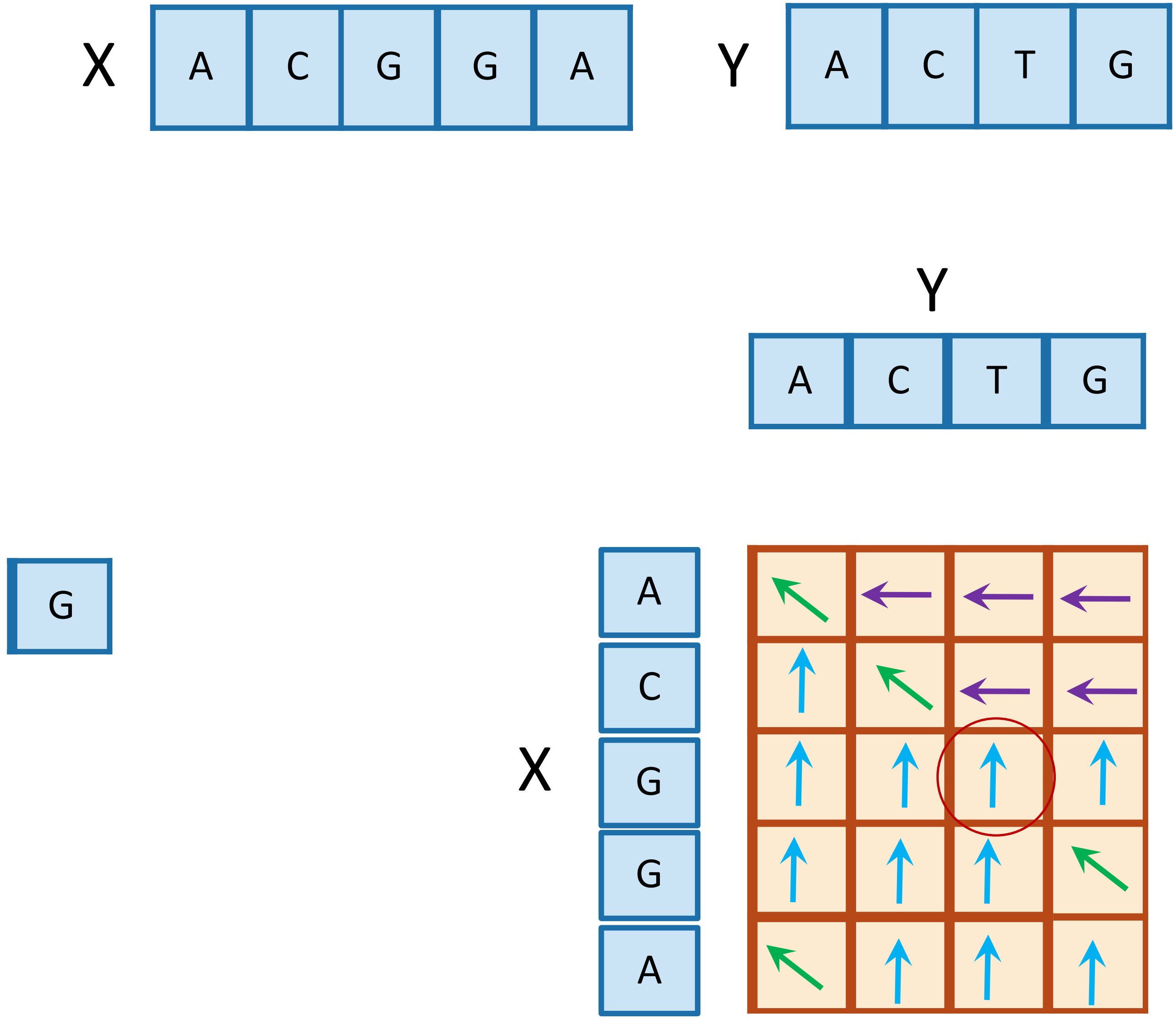


Table B

Example

X | A | C | G | G | A

Y | A | C | T | G

G

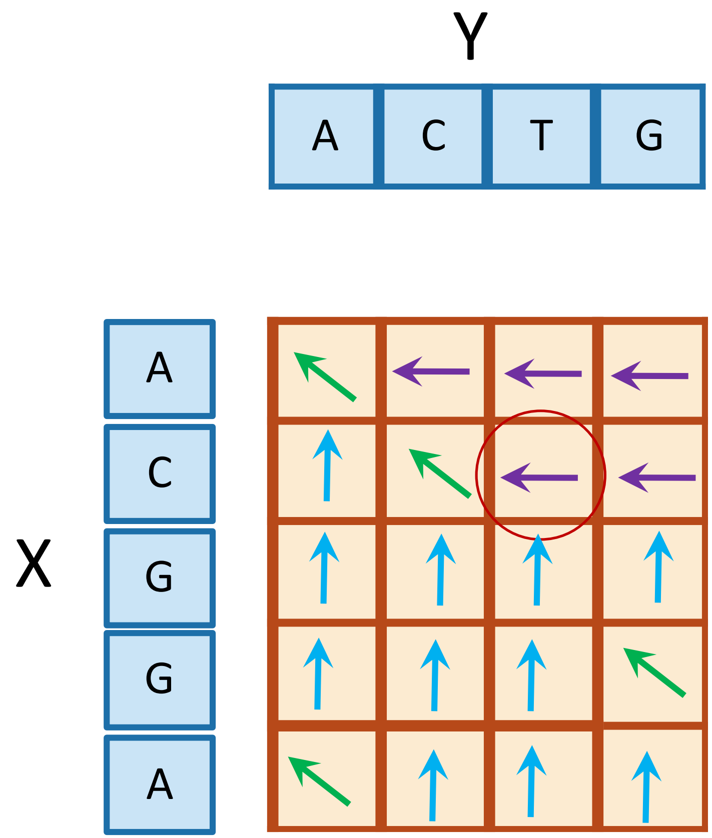
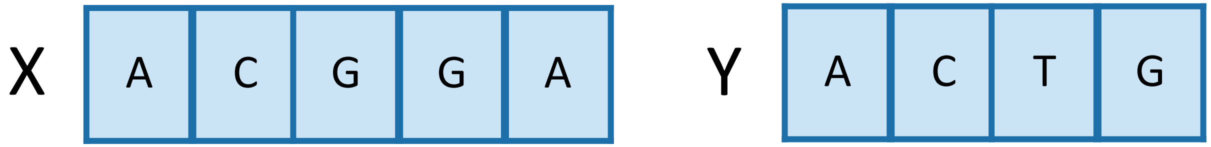
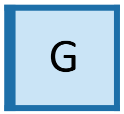


Table B

Example





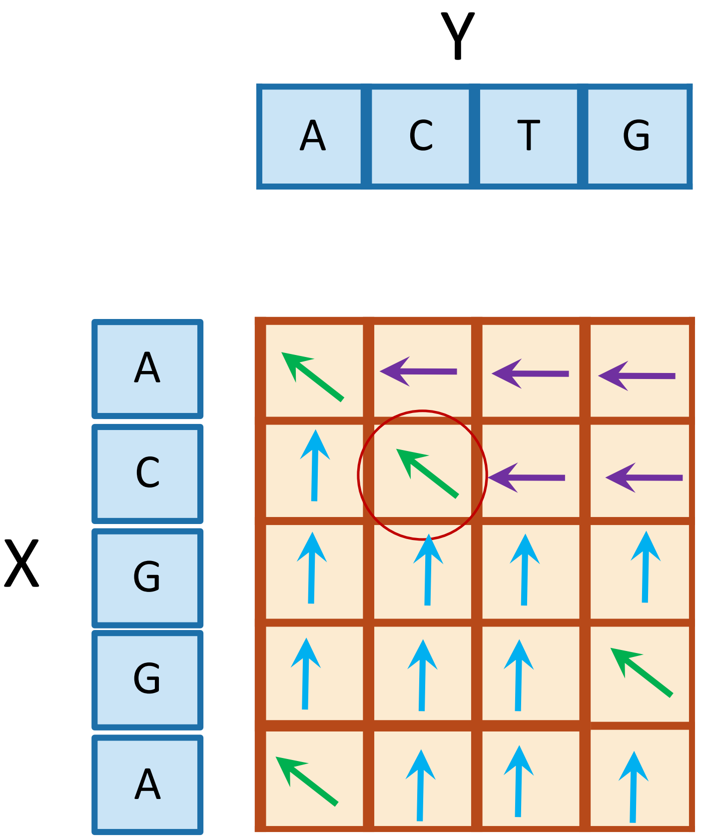


Table B

Example

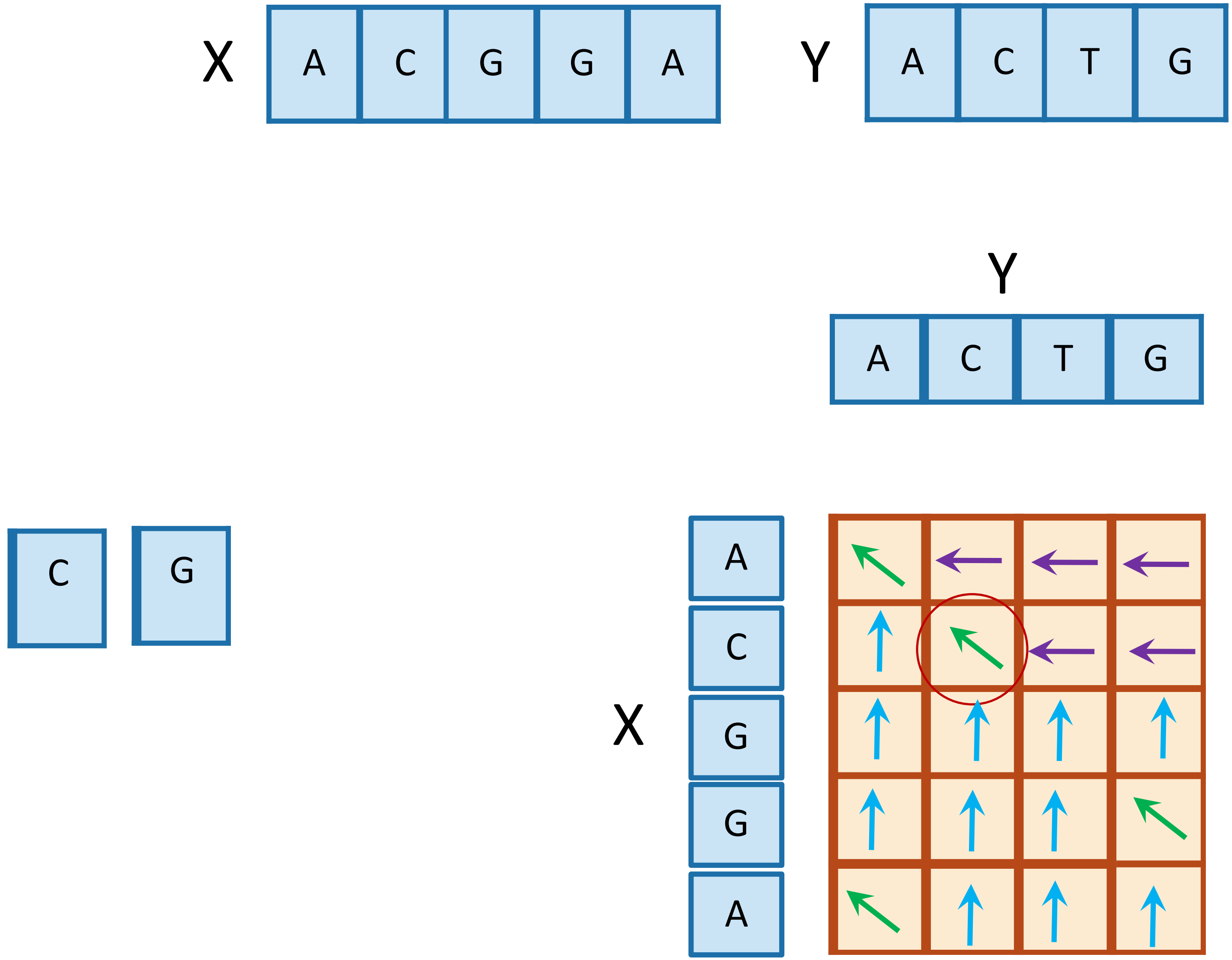
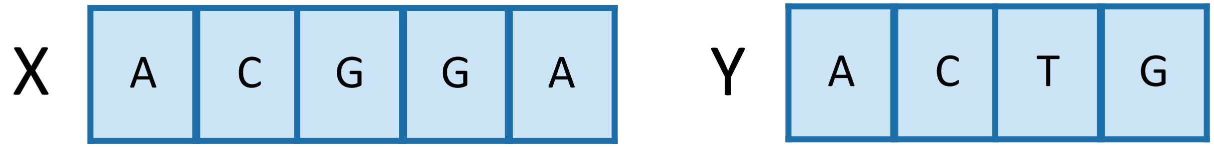
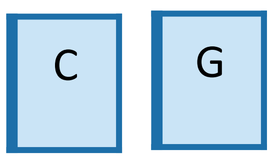


Table B

Example





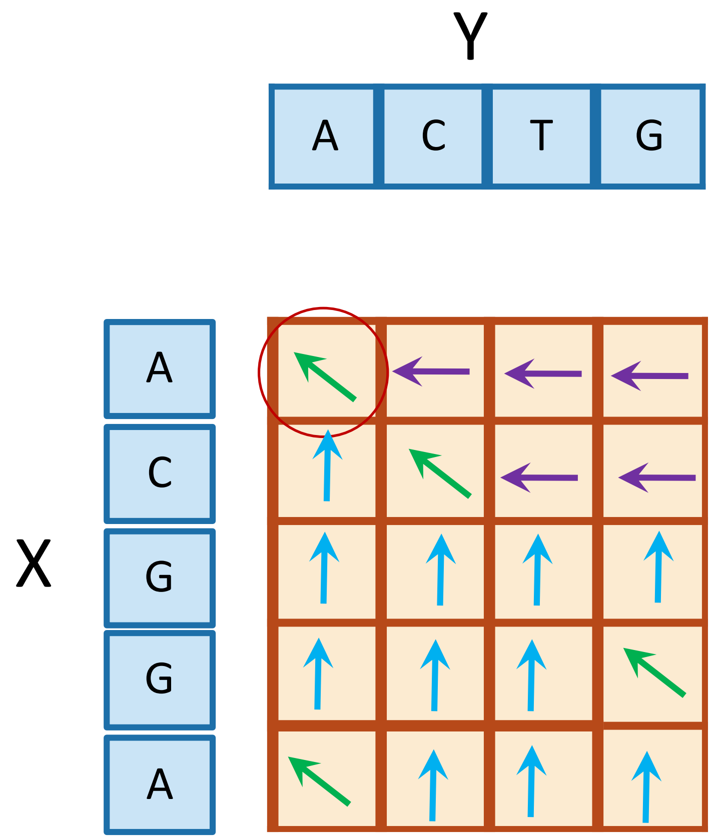
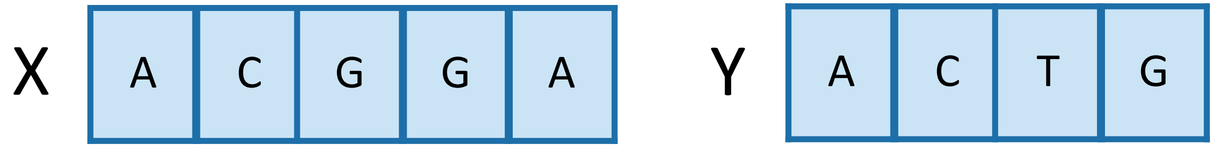
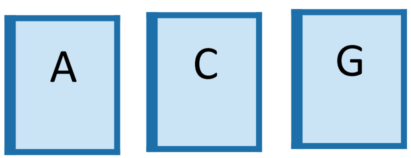


Table B

Example





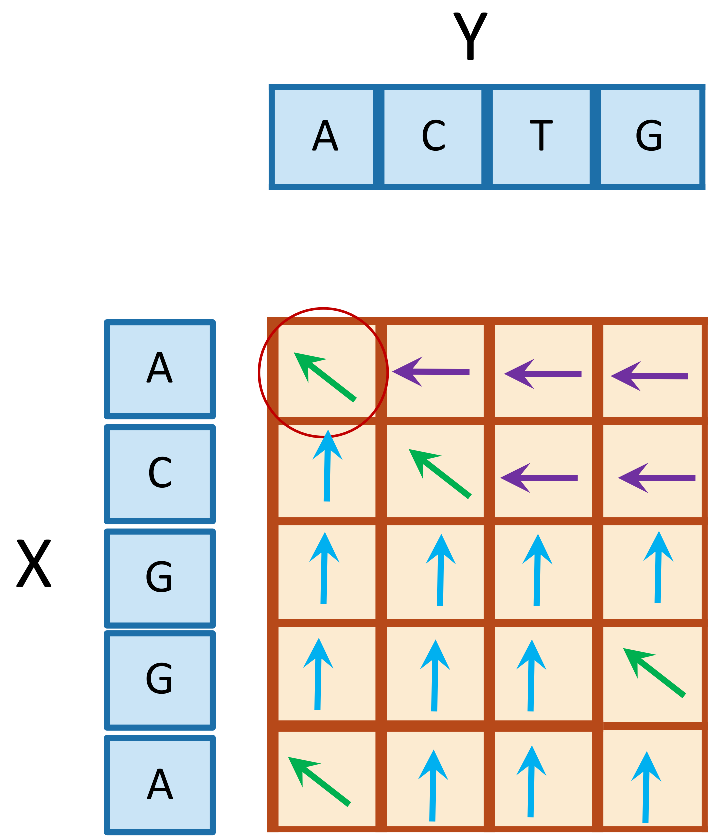


Table B

Example2

X

A	B	C	B	D	A	B

Y

B	D	C	A	B	A

Example2

X	A	B	C	B	D	A	B

Y	B	D	C	A	B	A

		B	D	C	A	B	A
A							
B							
C							
B							
D							
A							
B							

Table C

B	D	C	A	B	A

Table B

$$C[i,j] = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ C[i-1, j-1] + 1 & if\ X[i] = Y[j] and\ i, j > 0 \\ \max\{\ C[i, j-1], C[i, j-1]\} & if X[i] \neq Y[j] and\ i, j > 0 \end{cases}$$

Steps for applying Dynamic Programming

Step 4: If needed, keep track of some additional info so that the algorithm from Step 3 can find the actual LCS.

Algorithm

```
PRINT-LCS(B, X, i, j)
if i == 0 or j == 0
        return                          // the LCS has length 0
if B[i, j] == "↖"
        PRINT-LCS(B, X, i, −1 j − 1)
        print X[i]                      // same as Y[j]
elseif B[i, j] == "↑"
        PRINT-LCS(B, X, i, −1 j)
else
        PRINT-LCS(B, X, i, j − 1)
```