BIOS 477/877
Bioinformatics and Molecular Evolution

Instructor: Etsuko Moriyama
(School of Biological Sciences)

Spring 2026 **Lecture 6**

Today's topics

- Pairwise alignment
  - Dotplot
  - Simple alignments (scoring scheme)
  - Dynamic programming algorithm
- Assignment 3
  (15 points total, very short!)
  Due: February 1 (Sun)

Why compare sequences?

- What can we learn from comparing sequences?
  - Their relatedness:
    - Protein/gene family classification
    - Evolutionary relationships/history
  - Conserved sequence features
    - *e.g.*, functional domains/motifs
      ➔ possibly sharing the same functions

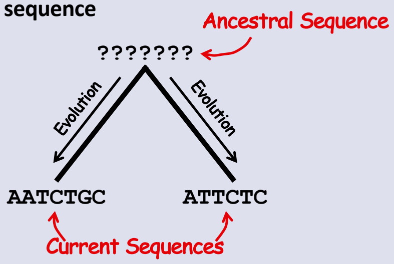
➡ Can be used to predict gene/protein functions and structures

Pairwise sequence comparison

- When we compare similar sequences:

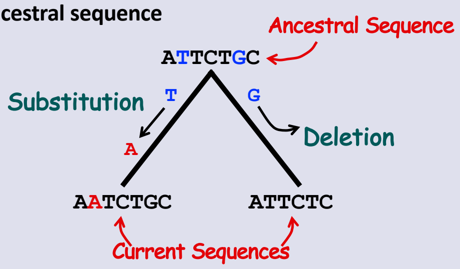
Pairwise sequence comparison:
hypothesis – common descent

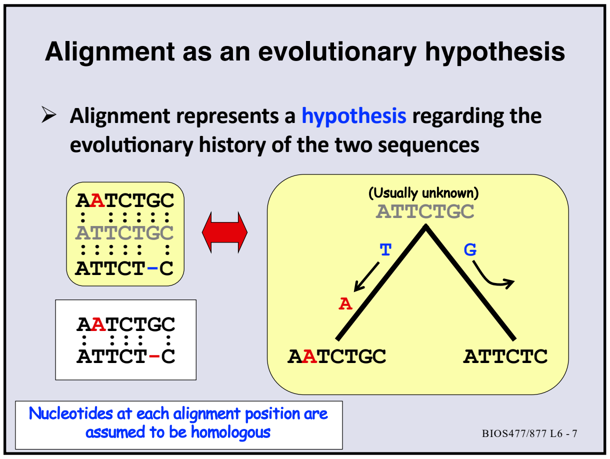
- Hypothesis: they are derived from a common ancestral sequence



Pairwise sequence comparison:
hypothesis – common descent (continued)

- Hypothesis: they are derived from a common ancestral sequence

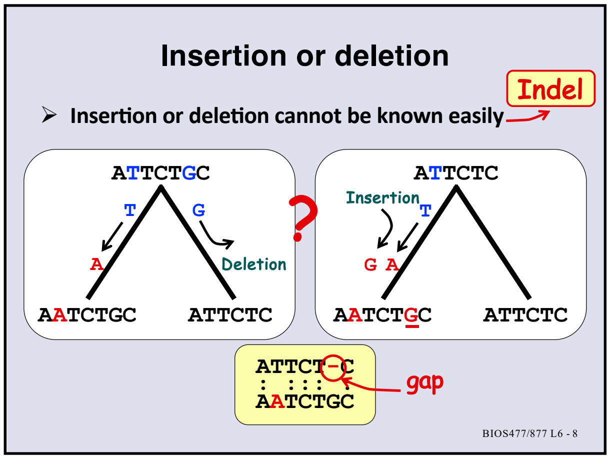




Alignment as an evolutionary hypothesis

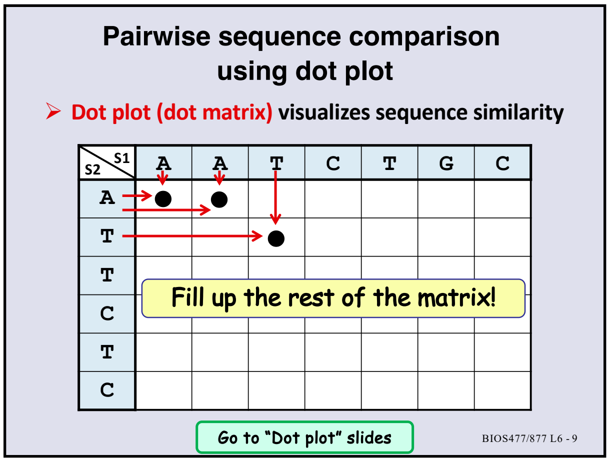
- Alignment represents a hypothesis regarding the evolutionary history of the two sequences

Nucleotides at each alignment position are assumed to be homologous



Insertion or deletion

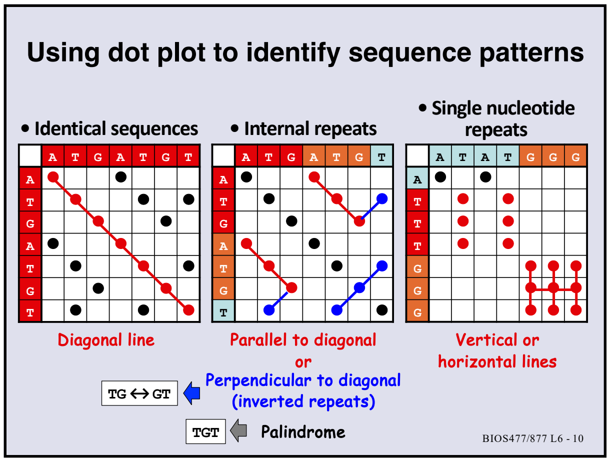
- Insertion or deletion cannot be known easily



Pairwise sequence comparison using dot plot

- Dot plot (dot matrix) visualizes sequence similarity

Go to "Dot plot" slides



Using dot plot to identify sequence patterns

- Identical sequences
- Internal repeats
- Single nucleotide repeats

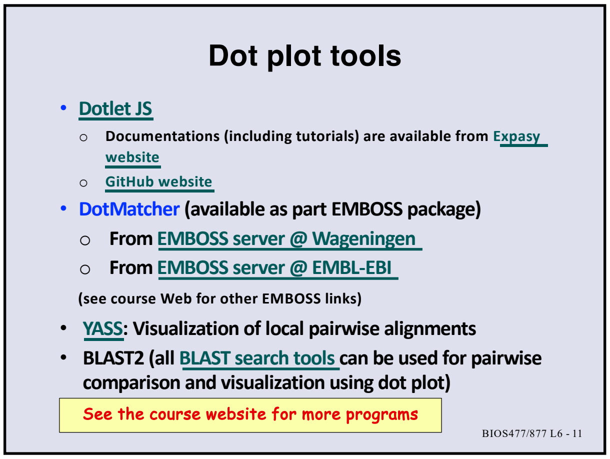
Diagonal line

Parallel to diagonal
or
Perpendicular to diagonal
(inverted repeats)

Vertical or horizontal lines

TG ↔ GT

TGT Palindrome

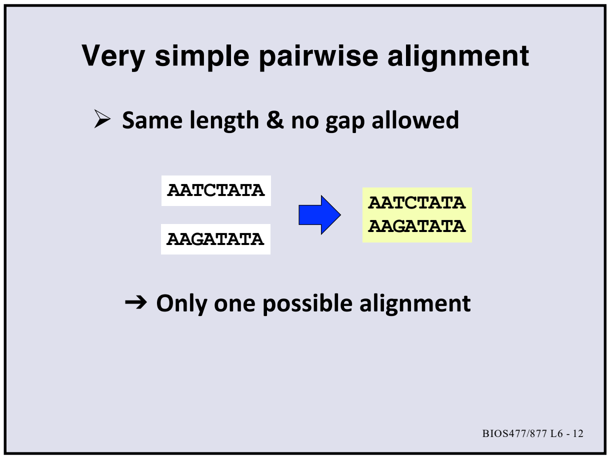


Dot plot tools

- Dotlet JS
  - Documentations (including tutorials) are available from Expasy website
  - GitHub website
- DotMatcher (available as part EMBOSS package)
  - From EMBOSS server @ Wagengingen
  - From EMBOSS server @ EMBL-EBI

  (see course Web for other EMBOSS links)
- YASS: Visualization of local pairwise alignments
- BLAST2 (all BLAST search tools can be used for pairwise comparison and visualization using dot plot)

See the course website for more programs



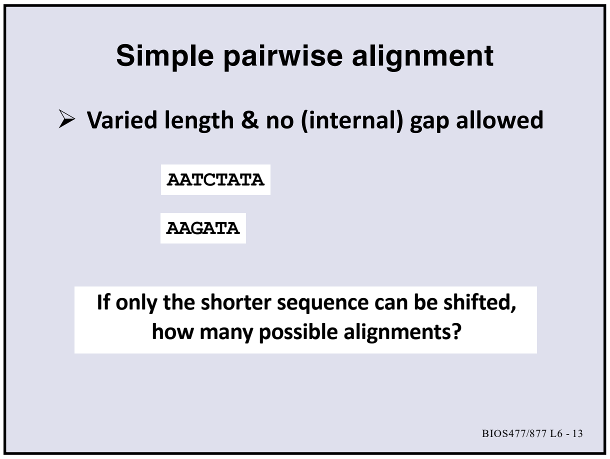
Very simple pairwise alignment

- Same length & no gap allowed

AATCTATA
AAGATATA

AATCTATA
AAGATATA

→ Only one possible alignment



Simple pairwise alignment

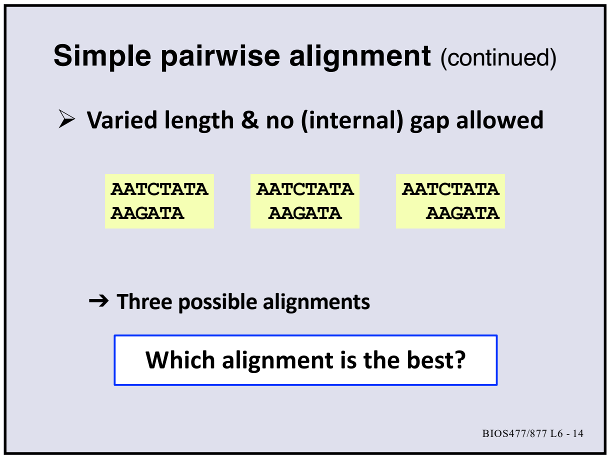
- ➢ Varied length & no (internal) gap allowed

```
AATCTATA

AAGATA
```

If only the shorter sequence can be shifted, how many possible alignments?

13



Simple pairwise alignment (continued)

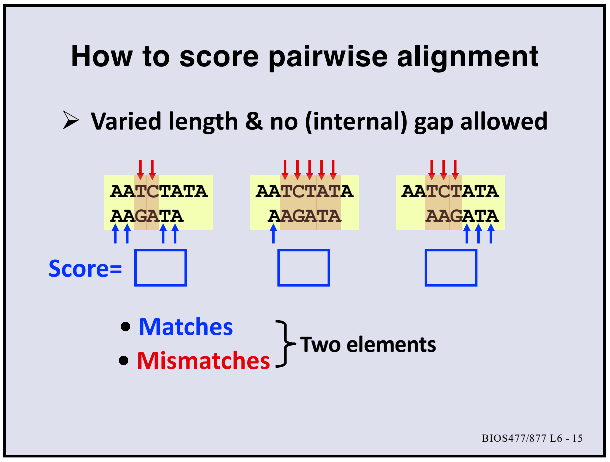
- ➢ Varied length & no (internal) gap allowed

```
AATCTATA    AATCTATA    AATCTATA
AAGATA       AAGATA       AAGATA
```

➔ Three possible alignments

Which alignment is the best?

14



How to score pairwise alignment

- ➢ Varied length & no (internal) gap allowed

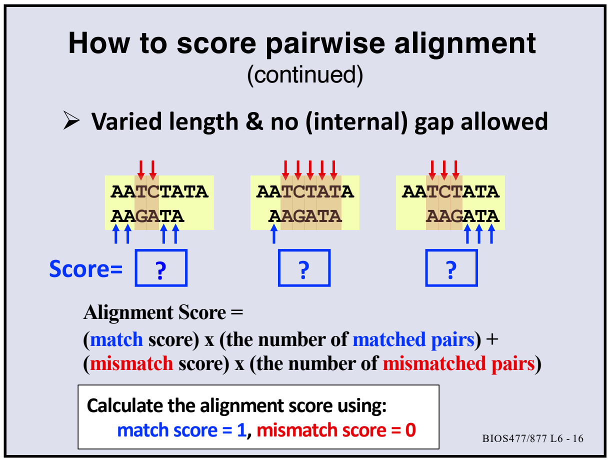
```
  ↓↓          ↓↓↓↓↓        ↓↓↓
AATCTATA    AATCTATA    AATCTATA
AAGATA       AAGATA       AAGATA
↑↑  ↑↑       ↑               ↑↑↑
```

Score=

- **Matches**
- **Mismatches**

} Two elements

15



How to score pairwise alignment
(continued)

- ➢ Varied length & no (internal) gap allowed

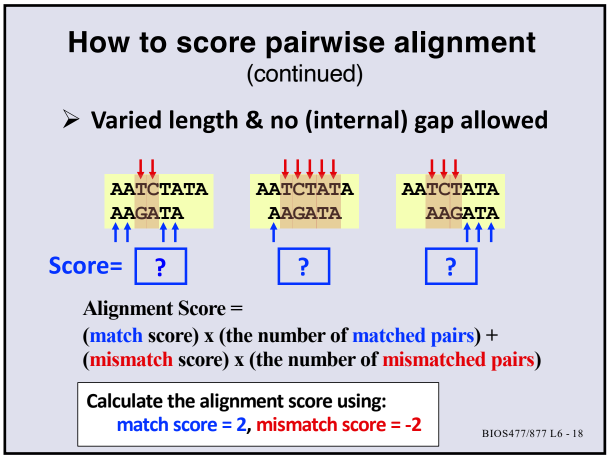
```
  ↓↓          ↓↓↓↓↓        ↓↓↓
AATCTATA    AATCTATA    AATCTATA
AAGATA       AAGATA       AAGATA
↑↑  ↑↑       ↑               ↑↑↑
```

Score= ? ? ?

Alignment Score =
(match score) x (the number of matched pairs) +
(mismatch score) x (the number of mismatched pairs)

Calculate the alignment score using:
match score = 1, mismatch score = 0

16



How to score pairwise alignment
(continued)

- ➢ Varied length & no (internal) gap allowed

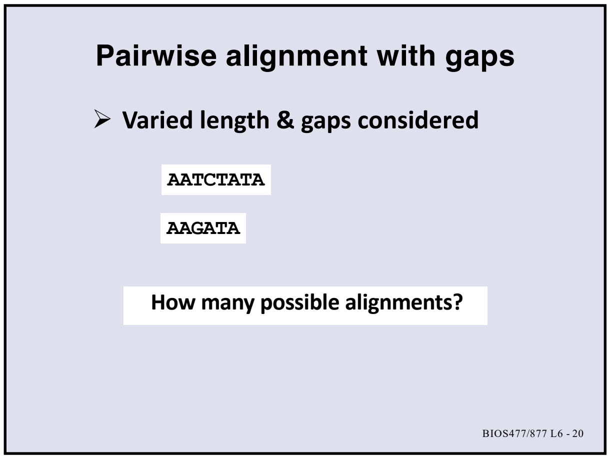
```
  ↓↓          ↓↓↓↓↓        ↓↓↓
AATCTATA    AATCTATA    AATCTATA
AAGATA       AAGATA       AAGATA
↑↑  ↑↑       ↑               ↑↑↑
```

Score= ? ? ?

Alignment Score =
(match score) x (the number of matched pairs) +
(mismatch score) x (the number of mismatched pairs)

Calculate the alignment score using:
match score = 2, mismatch score = -2

18



Pairwise alignment with gaps

- ➢ Varied length & gaps considered

```
AATCTATA

AAGATA
```

How many possible alignments?

20

Pairwise alignment with gaps

- ➢ Varied length & gaps considered

AATCTATA
AA-G-ATA

AATCTATA
AA--GATA

AATCTATA
AAG-AT-A

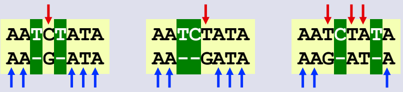
And many more ...

If two gaps are allowed in the shorter sequence ...

→ 28 alignments possible!
(3 above and 25 more)

Scoring pairwise alignment with gaps

- ➢ Varied length & gaps considered

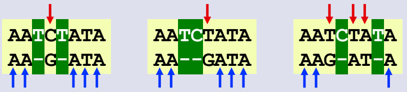


- **Matches**
- **Mismatches**
- **Gaps**

} Three elements

Scoring pairwise alignment with gaps
(continued)

- ➢ Varied length & gaps considered



Score= ? ? ?

Alignment Score =
(match score) x (the number of matched pairs) +
(mismatch score) x (the number of mismatched pairs) +
(gap penalty) x (the number of gaps)

Use match score = 1, mismatch score = 0, gap penalty = -1

Pairwise alignment as an optimization problem

- ➢ To find the best alignment(s) between two sequences
  - ▪ Exhaustive search method
    - ➔ Examine all possible alignments
    - ➔ Develop a method to score each alignment (Match score, mismatch score, gap penalty)
    - ➔ Choose the best (optimal) alignment based on the score
    - ➔ Not feasible for long sequences (e.g., 100bp x 95bp with 5 gaps only in the shorter sequence would require testing ~ 55 million possible alignments!)

Need more efficient strategy!

Dynamic programming algorithm:
an optimization method

- ➢ What is dynamic programming?
  - ➔ An algorithm in which an optimization problem is solved by saving the optimal scores for the solution of every sub-problem instead of recalculating them repetitively.
    - A problem is broken apart into reasonably sized sub-problems.
    - These partial results are used to compute the final answer.

It is used in many bioinformatics methods

Dynamic programming algorithm for pairwise alignment

- ➢ Needleman & Wunsch (1970) applied the dynamic programming approach to the sequence alignment problem.

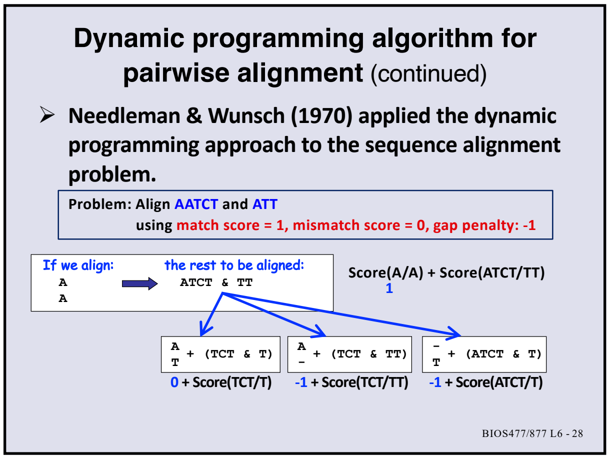
Problem: Align AATCT and ATT
using match score = 1, mismatch score = 0, gap penalty: -1

If we align: A / A ➔ *the rest to be aligned:* ATCT & TT
Score(A/A) + Score(ATCT/TT)
1

If we align: A / - ➔ *the rest to be aligned:* ATCT & ATT
Score(A/-) + Score(ATCT/ATT)
-1

If we align: - / A ➔ *the rest to be aligned:* AATCT & TT
Score(-/A) + Score(AATCT/TT)
-1

Subalignment



Dynamic programming algorithm for pairwise alignment (continued)

- Needleman & Wunsch (1970) applied the dynamic programming approach to the sequence alignment problem.

Problem: Align AATCT and ATT
using match score = 1, mismatch score = 0, gap penalty: -1

If we align: A / A → the rest to be aligned: ATCT & TT

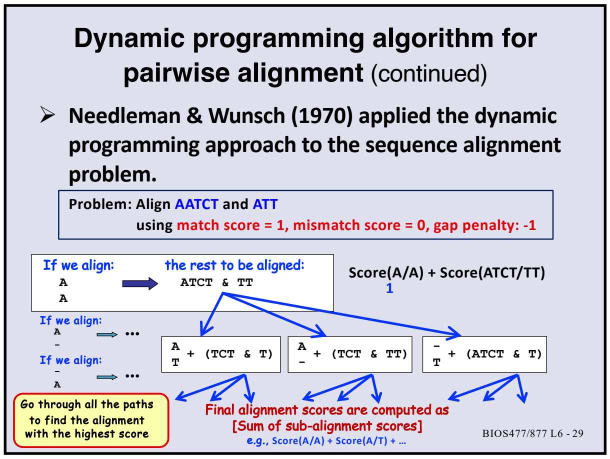
Score(A/A) + Score(ATCT/TT)
1

A / T + (TCT & T)
A / - + (TCT & TT)
- / T + (ATCT & T)

0 + Score(TCT/T)
-1 + Score(TCT/TT)
-1 + Score(ATCT/T)

BIOS477/877 L6 - 28

28



Dynamic programming algorithm for pairwise alignment (continued)

- Needleman & Wunsch (1970) applied the dynamic programming approach to the sequence alignment problem.

Problem: Align AATCT and ATT
using match score = 1, mismatch score = 0, gap penalty: -1

If we align: A / A → the rest to be aligned: ATCT & TT

Score(A/A) + Score(ATCT/TT)
1

If we align: A / - → •••

If we align: - / A → •••

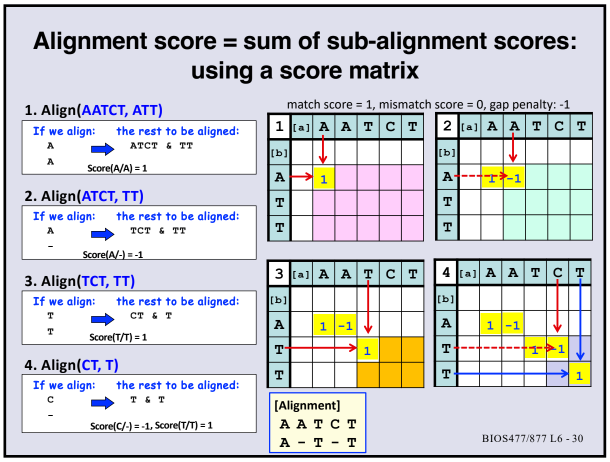
A / T + (TCT & T)
A / - + (TCT & TT)
- / T + (ATCT & T)

Go through all the paths to find the alignment with the highest score

Final alignment scores are computed as [Sum of sub-alignment scores]
e.g., Score(A/A) + Score(A/T) + …

BIOS477/877 L6 - 29

29



Alignment score = sum of sub-alignment scores: using a score matrix

1. Align(AATCT, ATT)

If we align: A / A → the rest to be aligned: ATCT & TT
Score(A/A) = 1

2. Align(ATCT, TT)

If we align: A / - → the rest to be aligned: TCT & TT
Score(A/-) = -1

3. Align(TCT, TT)

If we align: T / T → the rest to be aligned: CT & T
Score(T/T) = 1

4. Align(CT, T)

If we align: C / - → the rest to be aligned: T & T
Score(C/-) = -1, Score(T/T) = 1

match score = 1, mismatch score = 0, gap penalty: -1

1	[a]	A	A	T	C	T
[b]						
A		1				
T						
T						

2	[a]	A	A	T	C	T
[b]						
A		1	-1			
T						
T						

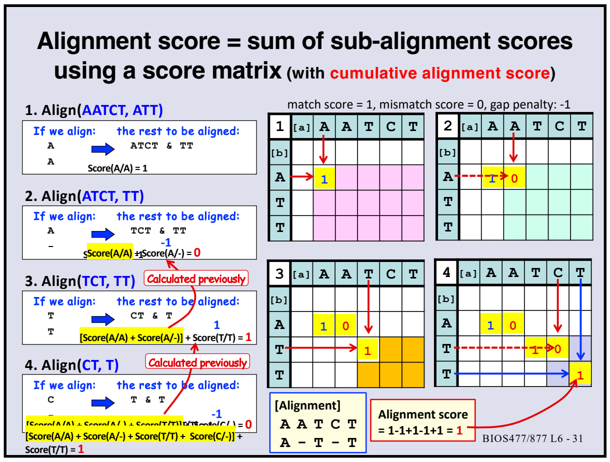
3	[a]	A	A	T	C	T
[b]						
A		1	-1			
T				1		
T						

4	[a]	A	A	T	C	T
[b]						
A		1	-1			
T				1	-1	
T						1

[Alignment]
A A T C T
A - T - T

BIOS477/877 L6 - 30

30



Alignment score = sum of sub-alignment scores using a score matrix (with cumulative alignment score)

1. Align(AATCT, ATT)

If we align: A / A → the rest to be aligned: ATCT & TT
Score(A/A) = 1

2. Align(ATCT, TT)

If we align: A / - → the rest to be aligned: TCT & TT
-1
Score(A/A) + Score(A/-) = 0

3. Align(TCT, TT) Calculated previously

If we align: T / T → the rest to be aligned: CT & T
1
[Score(A/A) + Score(A/-)] + Score(T/T) = 1

4. Align(CT, T) Calculated previously

If we align: C / - → the rest to be aligned: T & T
-1
[Score(A/A) + Score(A/-) + Score(T/T)] + Score(C/-) = 0
[Score(A/A) + Score(A/-) + Score(T/T) + Score(C/-)] + Score(T/T) = 1

match score = 1, mismatch score = 0, gap penalty: -1

1	[a]	A	A	T	C	T
[b]						
A		1				
T						
T						

2	[a]	A	A	T	C	T
[b]						
A		1	0			
T						
T						

3	[a]	A	A	T	C	T
[b]						
A		1	0			
T				1		
T						

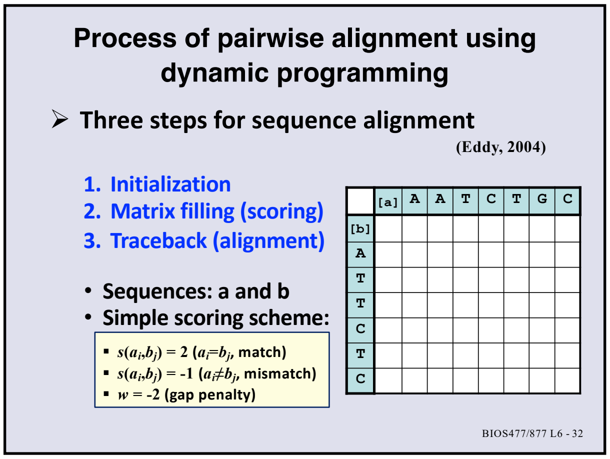
4	[a]	A	A	T	C	T
[b]						
A		1	0			
T				1	0	
T						1

[Alignment]
A A T C T
A - T - T

Alignment score = 1-1+1-1+1 = 1

BIOS477/877 L6 - 31

31



Process of pairwise alignment using dynamic programming

- Three steps for sequence alignment

(Eddy, 2004)

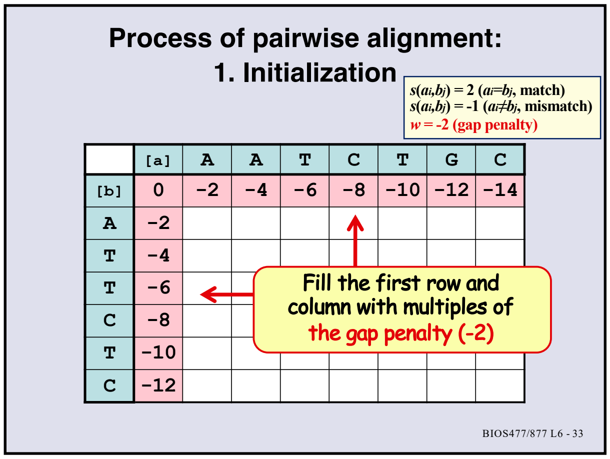
1. Initialization
2. Matrix filling (scoring)
3. Traceback (alignment)

- Sequences: a and b
- Simple scoring scheme:
  - $s(a_i,b_j) = 2$ ($a_i=b_j$, match)
  - $s(a_i,b_j) = -1$ ($a_i \neq b_j$, mismatch)
  - $w = -2$ (gap penalty)

	[a]	A	A	T	C	T	G	C
[b]								
A								
T								
T								
C								
T								
C								

BIOS477/877 L6 - 32

32



Process of pairwise alignment: 1. Initialization

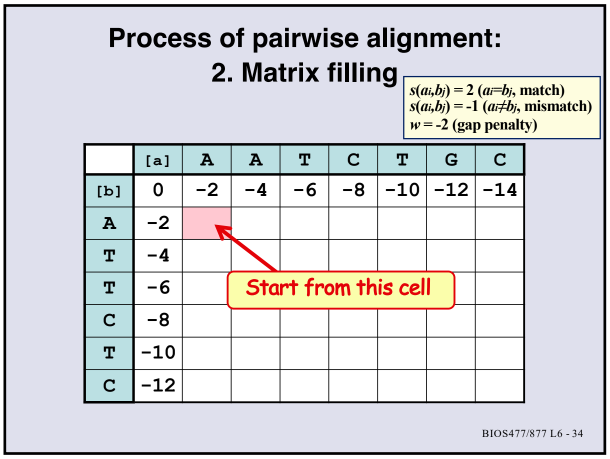
$s(a_i,b_j) = 2$ ($a_i=b_j$, match)
$s(a_i,b_j) = -1$ ($a_i \neq b_j$, mismatch)
$w = -2$ (gap penalty)

	[a]	A	A	T	C	T	G	C
[b]	0	-2	-4	-6	-8	-10	-12	-14
A	-2							
T	-4							
T	-6							
C	-8							
T	-10							
C	-12							

Fill the first row and column with multiples of the gap penalty (-2)

BIOS477/877 L6 - 33

33

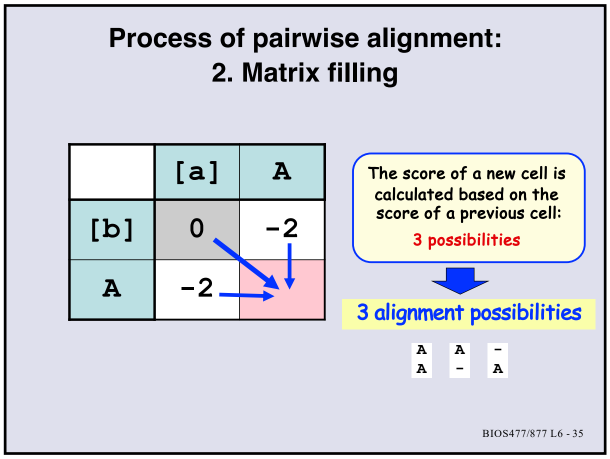


Process of pairwise alignment: 2. Matrix filling

$s(a_i,b_j) = 2$ ($a_i=b_j$, match)
$s(a_i,b_j) = -1$ ($a_i \neq b_j$, mismatch)
$w = -2$ (gap penalty)

	[a]	A	A	T	C	T	G	C
[b]	0	-2	-4	-6	-8	-10	-12	-14
A	-2							
T	-4							
T	-6							
C	-8							
T	-10							
C	-12							

Start from this cell



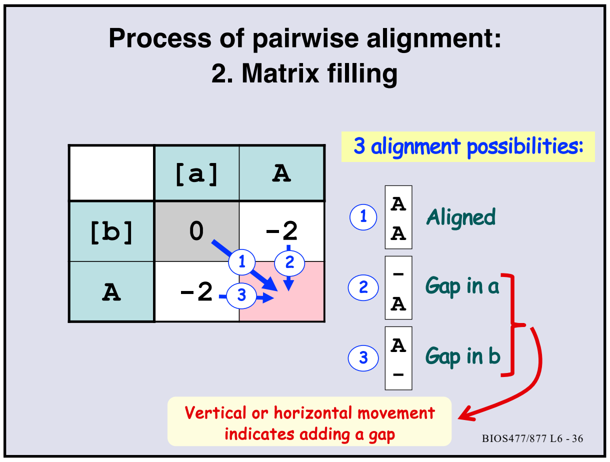
Process of pairwise alignment: 2. Matrix filling

	[a]	A
[b]	0	-2
A	-2	

The score of a new cell is calculated based on the score of a previous cell:
3 possibilities

3 alignment possibilities

A	A	-
A	-	A



Process of pairwise alignment: 2. Matrix filling

	[a]	A
[b]	0	-2
A	-2	

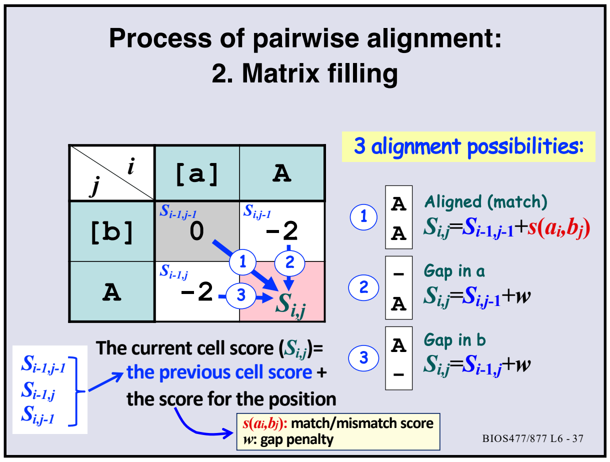
3 alignment possibilities:

① A / A Aligned

② - / A Gap in a

③ A / - Gap in b

Vertical or horizontal movement indicates adding a gap



Process of pairwise alignment: 2. Matrix filling

$j \backslash i$	[a]	A
[b]	$S_{i-1,j-1}$ 0	$S_{i,j-1}$ -2
A	$S_{i-1,j}$ -2	$S_{i,j}$

3 alignment possibilities:

① A / A Aligned (match) $S_{i,j}=S_{i-1,j-1}+s(a_i,b_j)$

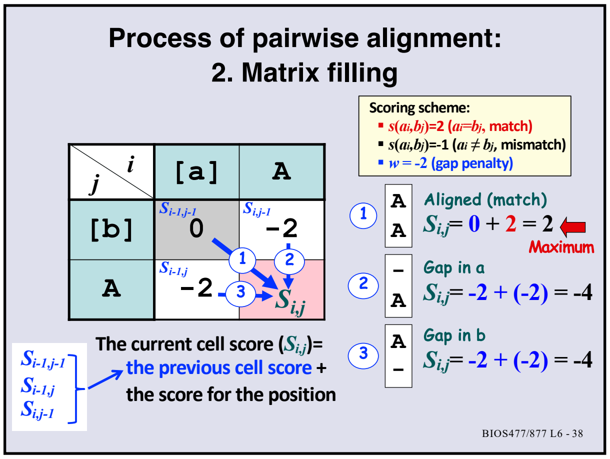
② - / A Gap in a $S_{i,j}=S_{i,j-1}+w$

③ A / - Gap in b $S_{i,j}=S_{i-1,j}+w$

The current cell score ($S_{i,j}$)=
the previous cell score +
the score for the position

$S_{i-1,j-1}$
$S_{i-1,j}$
$S_{i,j-1}$

$s(a_i,b_j)$: match/mismatch score
w: gap penalty



Process of pairwise alignment: 2. Matrix filling

Scoring scheme:
- $s(a_i,b_j)=2$ ($a_i=b_j$, match)
- $s(a_i,b_j)=-1$ ($a_i \neq b_j$, mismatch)
- $w = -2$ (gap penalty)

$j \backslash i$	[a]	A
[b]	$S_{i-1,j-1}$ 0	$S_{i,j-1}$ -2
A	$S_{i-1,j}$ -2	$S_{i,j}$

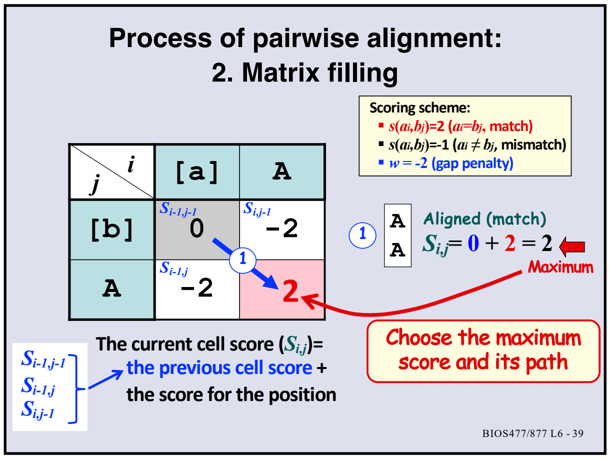
① A / A Aligned (match) $S_{i,j}= 0 + 2 = 2$ ⇐ Maximum

② - / A Gap in a $S_{i,j}= -2 + (-2) = -4$

③ A / - Gap in b $S_{i,j}= -2 + (-2) = -4$

The current cell score ($S_{i,j}$)=
the previous cell score +
the score for the position

$S_{i-1,j-1}$
$S_{i-1,j}$
$S_{i,j-1}$



Process of pairwise alignment: 2. Matrix filling

Scoring scheme:
- $s(a_i,b_j)=2$ ($a_i=b_j$, match)
- $s(a_i,b_j)=-1$ ($a_i \neq b_j$, mismatch)
- $w = -2$ (gap penalty)

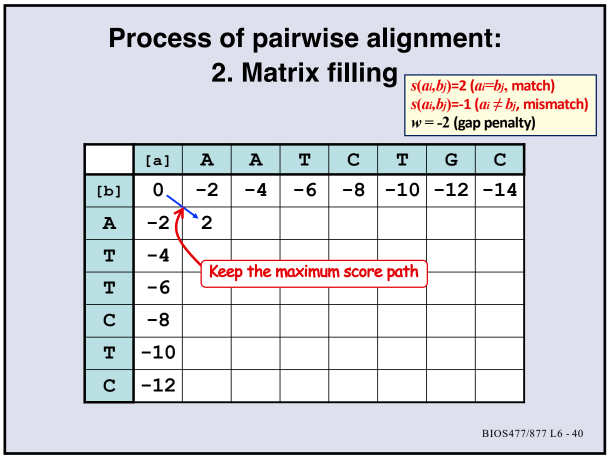
$j \backslash i$	[a]	A
[b]	$S_{i-1,j-1}$ 0	$S_{i,j-1}$ -2
A	$S_{i-1,j}$ -2	2

① A / A Aligned (match) $S_{i,j}= 0 + 2 = 2$ ⇐ Maximum

Choose the maximum score and its path

The current cell score ($S_{i,j}$)=
the previous cell score +
the score for the position

$S_{i-1,j-1}$
$S_{i-1,j}$
$S_{i,j-1}$

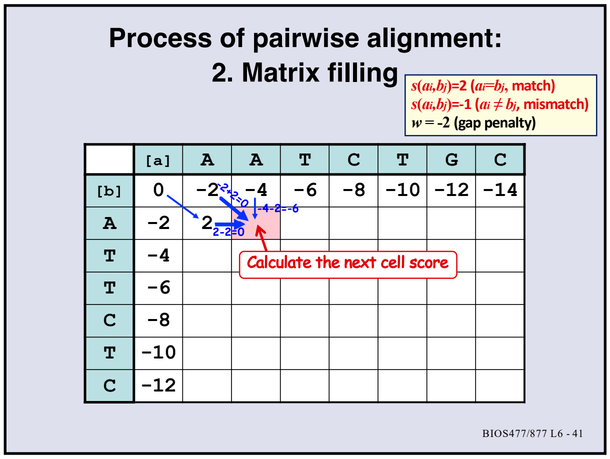


Process of pairwise alignment: 2. Matrix filling

$s(a_i,b_j)=2$ ($a_i=b_j$, match)
$s(a_i,b_j)=-1$ ($a_i \neq b_j$, mismatch)
$w=-2$ (gap penalty)

	[a]	A	A	T	C	T	G	C
[b]	0	−2	−4	−6	−8	−10	−12	−14
A	−2	2						
T	−4							
T	−6							
C	−8							
T	−10							
C	−12							

Keep the maximum score path

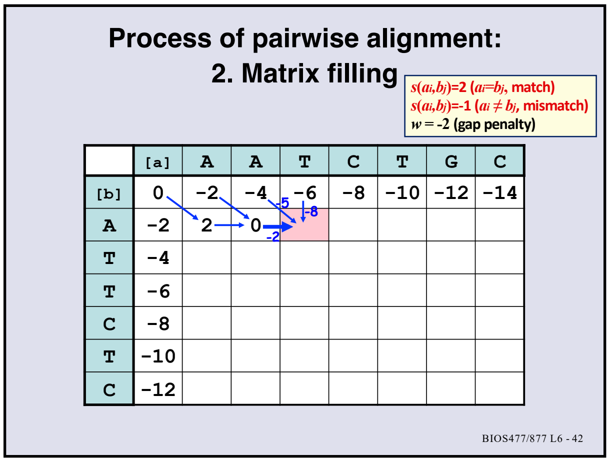


Process of pairwise alignment: 2. Matrix filling

$s(a_i,b_j)=2$ ($a_i=b_j$, match)
$s(a_i,b_j)=-1$ ($a_i \neq b_j$, mismatch)
$w=-2$ (gap penalty)

	[a]	A	A	T	C	T	G	C
[b]	0	−2	−4	−6	−8	−10	−12	−14
A	−2	2						
T	−4							
T	−6							
C	−8							
T	−10							
C	−12							

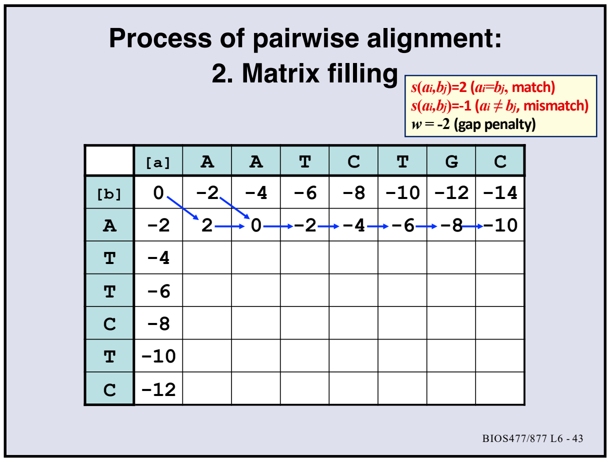
Calculate the next cell score



Process of pairwise alignment: 2. Matrix filling

$s(a_i,b_j)=2$ ($a_i=b_j$, match)
$s(a_i,b_j)=-1$ ($a_i \neq b_j$, mismatch)
$w=-2$ (gap penalty)

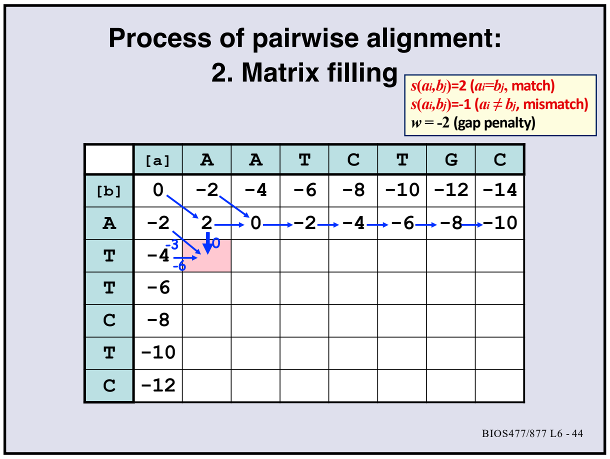
	[a]	A	A	T	C	T	G	C
[b]	0	−2	−4	−6	−8	−10	−12	−14
A	−2	2	0					
T	−4							
T	−6							
C	−8							
T	−10							
C	−12							



Process of pairwise alignment: 2. Matrix filling

$s(a_i,b_j)=2$ ($a_i=b_j$, match)
$s(a_i,b_j)=-1$ ($a_i \neq b_j$, mismatch)
$w=-2$ (gap penalty)

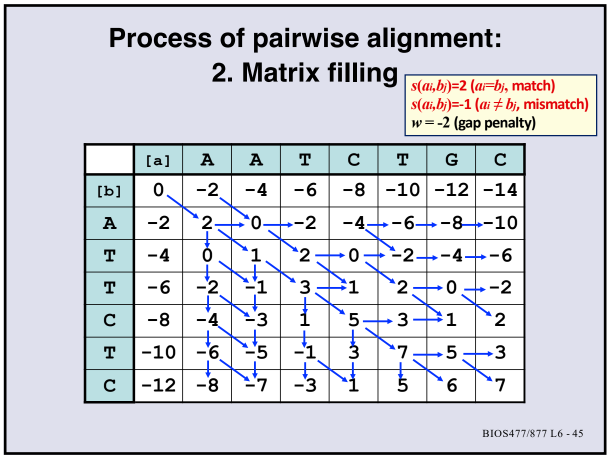
	[a]	A	A	T	C	T	G	C
[b]	0	−2	−4	−6	−8	−10	−12	−14
A	−2	2	0	−2	−4	−6	−8	−10
T	−4							
T	−6							
C	−8							
T	−10							
C	−12							



Process of pairwise alignment: 2. Matrix filling

$s(a_i,b_j)=2$ ($a_i=b_j$, match)
$s(a_i,b_j)=-1$ ($a_i \neq b_j$, mismatch)
$w=-2$ (gap penalty)

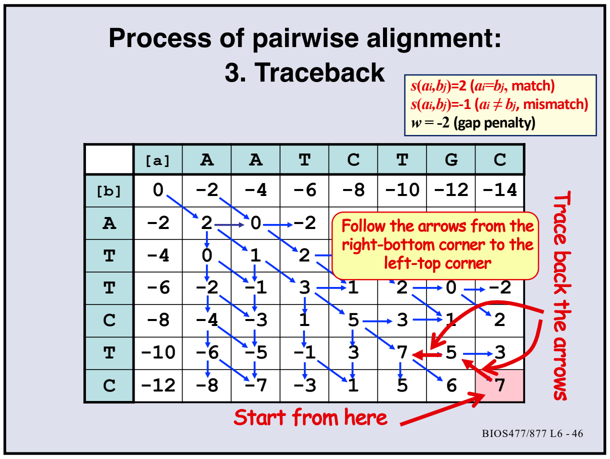
	[a]	A	A	T	C	T	G	C
[b]	0	−2	−4	−6	−8	−10	−12	−14
A	−2	2	0	−2	−4	−6	−8	−10
T	−4							
T	−6							
C	−8							
T	−10							
C	−12							



Process of pairwise alignment: 2. Matrix filling

$s(a_i,b_j)=2$ ($a_i=b_j$, match)
$s(a_i,b_j)=-1$ ($a_i \neq b_j$, mismatch)
$w=-2$ (gap penalty)

	[a]	A	A	T	C	T	G	C
[b]	0	−2	−4	−6	−8	−10	−12	−14
A	−2	2	0	−2	−4	−6	−8	−10
T	−4	0	1	2	0	−2	−4	−6
T	−6	−2	−1	3	1	2	0	−2
C	−8	−4	−3	1	5	3	1	2
T	−10	−6	−5	−1	3	7	5	3
C	−12	−8	−7	−3	1	5	6	7



Process of pairwise alignment: 3. Traceback

$s(a_i,b_j)=2$ ($a_i=b_j$, match)
$s(a_i,b_j)=-1$ ($a_i \neq b_j$, mismatch)
$w=-2$ (gap penalty)

	[a]	A	A	T	C	T	G	C
[b]	0	-2	-4	-6	-8	-10	-12	-14
A	-2	2	0	-2				
T	-4	0	1	2				
T	-6	-2	-1	3	1	2	0	-2
C	-8	-4	-3	1	5	3	1	2
T	-10	-6	-5	-1	3	7	5	3
C	-12	-8	-7	-3	1	5	6	7

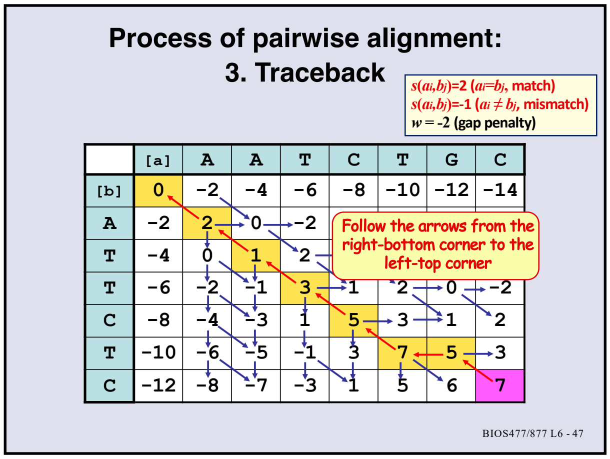
Follow the arrows from the right-bottom corner to the left-top corner

Trace back the arrows

Start from here

BIOS477/877 L6 - 46

46



Process of pairwise alignment: 3. Traceback

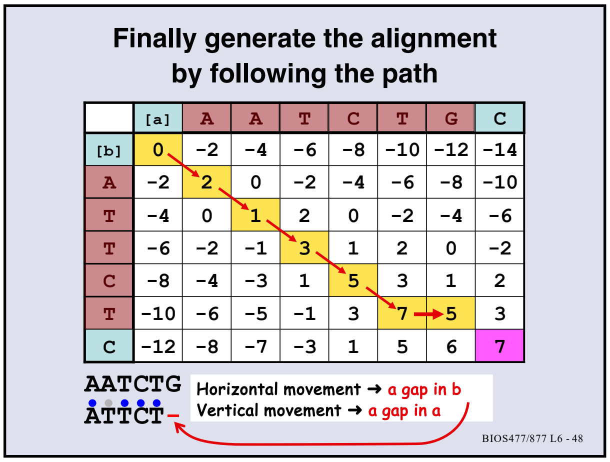
$s(a_i,b_j)=2$ ($a_i=b_j$, match)
$s(a_i,b_j)=-1$ ($a_i \neq b_j$, mismatch)
$w=-2$ (gap penalty)

	[a]	A	A	T	C	T	G	C
[b]	0	-2	-4	-6	-8	-10	-12	-14
A	-2	2	0	-2				
T	-4	0	1	2				
T	-6	-2	-1	3	1	2	0	-2
C	-8	-4	-3	1	5	3	1	2
T	-10	-6	-5	-1	3	7	5	3
C	-12	-8	-7	-3	1	5	6	7

Follow the arrows from the right-bottom corner to the left-top corner

BIOS477/877 L6 - 47

47



Finally generate the alignment by following the path

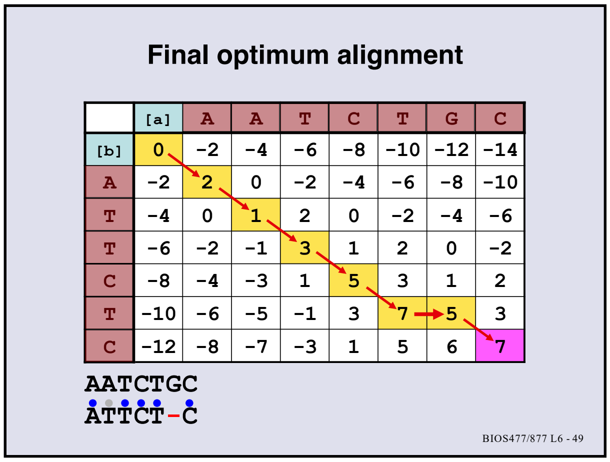
	[a]	A	A	T	C	T	G	C
[b]	0	-2	-4	-6	-8	-10	-12	-14
A	-2	2	0	-2	-4	-6	-8	-10
T	-4	0	1	2	0	-2	-4	-6
T	-6	-2	-1	3	1	2	0	-2
C	-8	-4	-3	1	5	3	1	2
T	-10	-6	-5	-1	3	7	5	3
C	-12	-8	-7	-3	1	5	6	7

```
AATCTG
ATTCT-
```

Horizontal movement ➔ a gap in b
Vertical movement ➔ a gap in a

BIOS477/877 L6 - 48

48



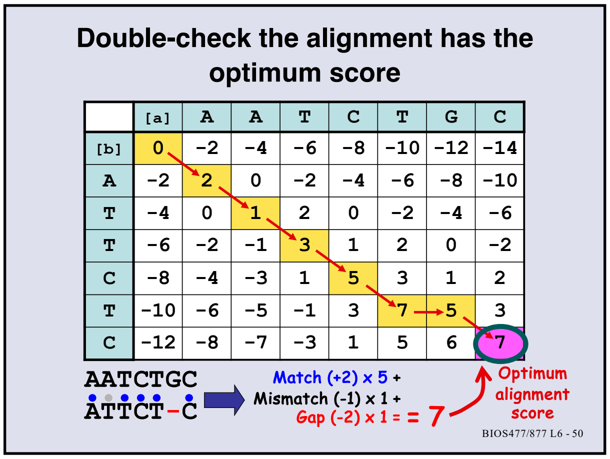
Final optimum alignment

	[a]	A	A	T	C	T	G	C
[b]	0	-2	-4	-6	-8	-10	-12	-14
A	-2	2	0	-2	-4	-6	-8	-10
T	-4	0	1	2	0	-2	-4	-6
T	-6	-2	-1	3	1	2	0	-2
C	-8	-4	-3	1	5	3	1	2
T	-10	-6	-5	-1	3	7	5	3
C	-12	-8	-7	-3	1	5	6	7

```
AATCTGC
ATTCT-C
```

BIOS477/877 L6 - 49

49



Double-check the alignment has the optimum score

	[a]	A	A	T	C	T	G	C
[b]	0	-2	-4	-6	-8	-10	-12	-14
A	-2	2	0	-2	-4	-6	-8	-10
T	-4	0	1	2	0	-2	-4	-6
T	-6	-2	-1	3	1	2	0	-2
C	-8	-4	-3	1	5	3	1	2
T	-10	-6	-5	-1	3	7	5	3
C	-12	-8	-7	-3	1	5	6	7

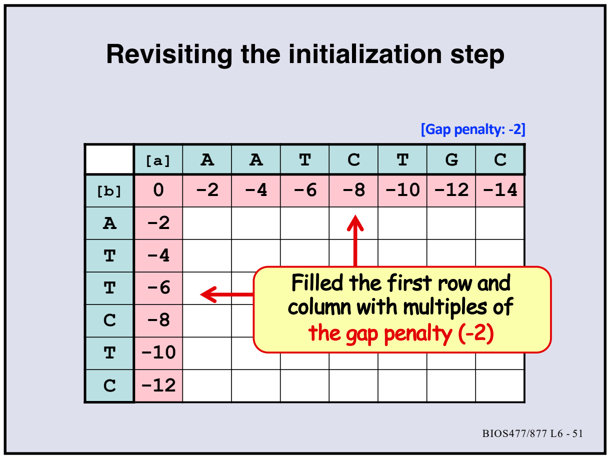
```
AATCTGC
ATTCT-C
```

Match (+2) x 5 +
Mismatch (-1) x 1 +
Gap (-2) x 1 = = 7

Optimum alignment score

BIOS477/877 L6 - 50

50



Revisiting the initialization step

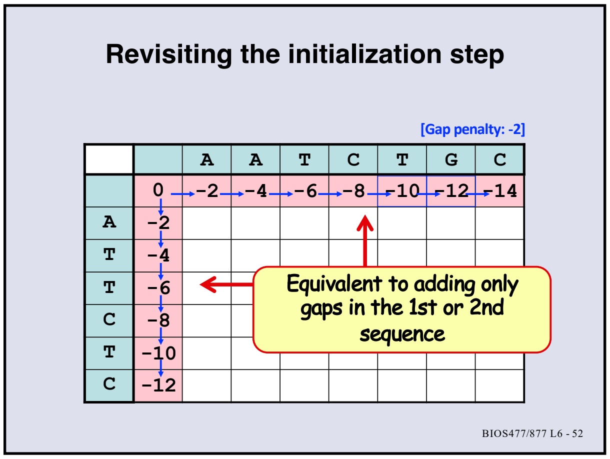
[Gap penalty: -2]

	[a]	A	A	T	C	T	G	C
[b]	0	-2	-4	-6	-8	-10	-12	-14
A	-2							
T	-4							
T	-6							
C	-8							
T	-10							
C	-12							

Filled the first row and column with multiples of the gap penalty (-2)

BIOS477/877 L6 - 51

51

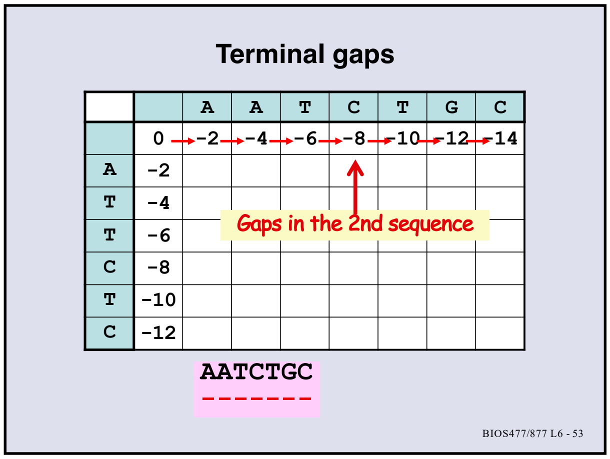


Revisiting the initialization step

[Gap penalty: -2]

		A	A	T	C	T	G	C
	0	-2	-4	-6	-8	-10	-12	-14
A	-2							
T	-4							
T	-6							
C	-8							
T	-10							
C	-12							

Equivalent to adding only gaps in the 1st or 2nd sequence

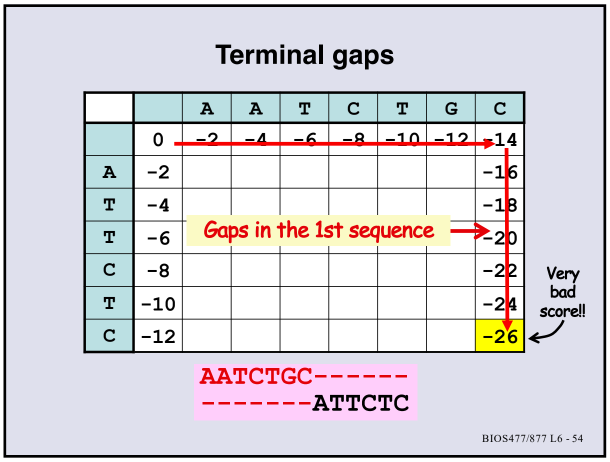


Terminal gaps

		A	A	T	C	T	G	C
	0	-2	-4	-6	-8	-10	-12	-14
A	-2							
T	-4							
T	-6							
C	-8							
T	-10							
C	-12							

Gaps in the 2nd sequence

AATCTGC



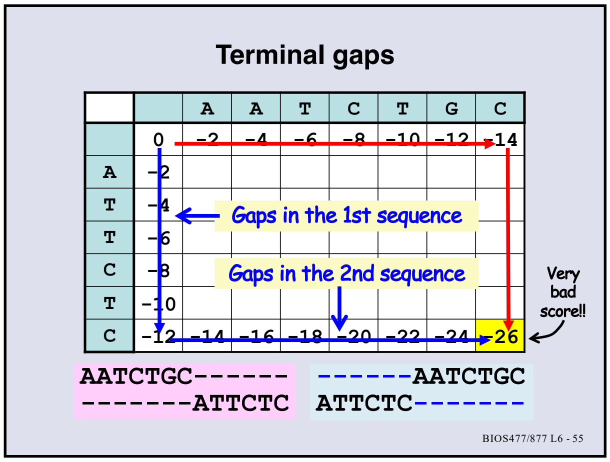
Terminal gaps

		A	A	T	C	T	G	C
	0	-2	-4	-6	-8	-10	-12	-14
A	-2							-16
T	-4							-18
T	-6							-20
C	-8							-22
T	-10							-24
C	-12							-26

Gaps in the 1st sequence

Very bad score!!

AATCTGC-------
-------ATTCTC



Terminal gaps

		A	A	T	C	T	G	C
	0	-2	-4	-6	-8	-10	-12	-14
A	-2							
T	-4							
T	-6							
C	-8							
T	-10							
C	-12	-14	-16	-18	-20	-22	-24	-26

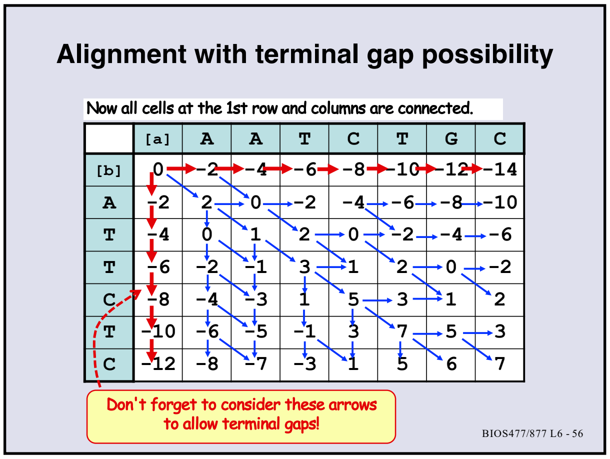
Gaps in the 1st sequence

Gaps in the 2nd sequence

Very bad score!!

AATCTGC------
-------ATTCTC

-------AATCTGC
ATTCTC-------

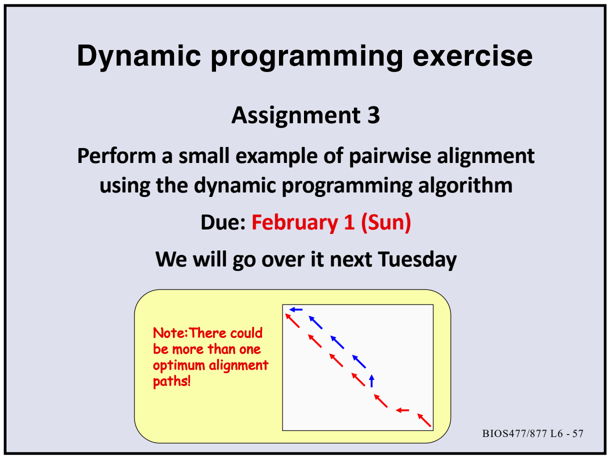


Alignment with terminal gap possibility

Now all cells at the 1st row and columns are connected.

	[a]	A	A	T	C	T	G	C
[b]	0	-2	-4	-6	-8	-10	-12	-14
A	-2	2	0	-2	-4	-6	-8	-10
T	-4	0	1	2	0	-2	-4	-6
T	-6	-2	-1	3	1	2	0	-2
C	-8	-4	-3	1	5	3	1	2
T	-10	-6	-5	-1	3	7	5	3
C	-12	-8	-7	-3	1	5	6	7

Don't forget to consider these arrows to allow terminal gaps!



Dynamic programming exercise

Assignment 3

Perform a small example of pairwise alignment using the dynamic programming algorithm

Due: February 1 (Sun)

We will go over it next Tuesday

Note: There could be more than one optimum alignment paths!