

# Lecture Notes for INF281 Basics of Bioinformatics Sequence Analysis

Takaya Saito



This work is licensed under a Creative Commons Attribution 4.0 International License.

# Contents

<b>I</b>	<b>1</b>	
<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Introduction to Molecular Biology . . . . .	1
1.2	Introduction to Biotechnology . . . . .	5
1.3	Bioinformatics . . . . .	6
 <b>II</b>		 <b>8</b>
<b>2</b>	<b>Global pairwise alignment</b>	<b>8</b>
2.1	Pairwise alignment . . . . .	8
2.2	Alignment by brute-force . . . . .	9
2.3	Table representation of alignment . . . . .	10
2.4	Global alignment with DP . . . . .	12
2.5	Backtracking . . . . .	14
2.6	Needleman-Wunsch algorithm . . . . .	17
<b>3</b>	<b>Extension of global alignment</b>	<b>18</b>
3.1	Introduction of score matrix . . . . .	18

# Part I

## 1 Introduction

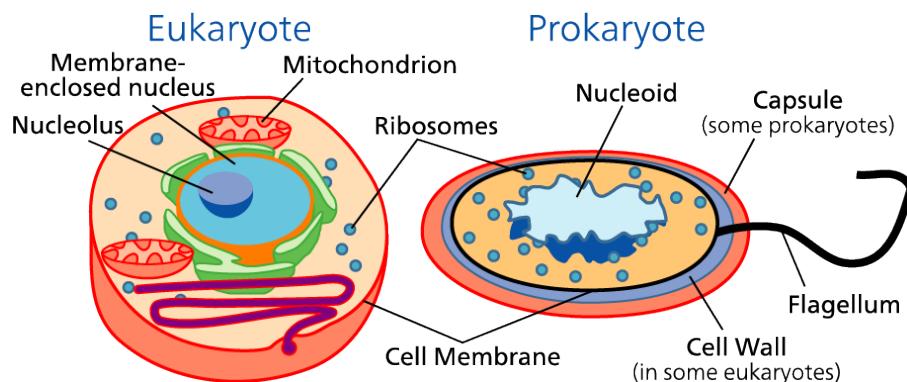
### 1.1 Introduction to Molecular Biology

Molecular biology is the study of biology focusing on organisms and cells at the molecular level.

#### Five essential facts about cells

##### 1. Two primary types of cells - eukaryotes and prokaryotes

- Eukaryote: animals & plants
- Prokaryote: bacteria & archaea



**Figure 1.1:** Eukaryotic and prokaryotic cells (source: Science Primer, Wikimedia Commons)

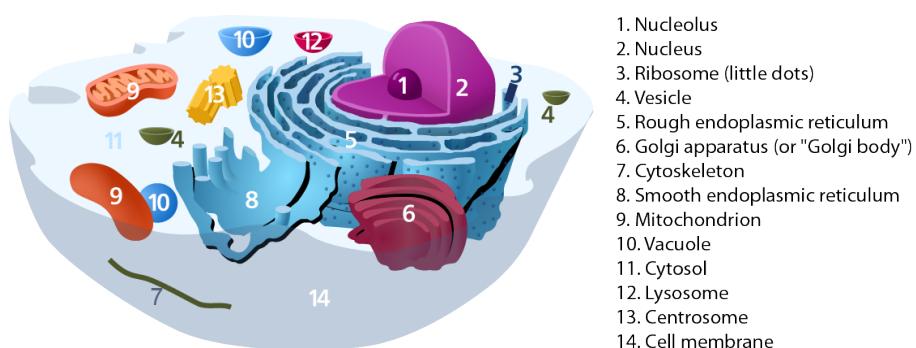
##### 2. Cell size - around 1 to 100 micrometers

- Cell Size and Scale: <http://learn.genetics.utah.edu/content/cells/scale>

##### 3. The number of cells

- Prokaryotes: 1 cell
- Human: Estimate of 15 trillion cells

##### 4. An animal cell and cell organelles



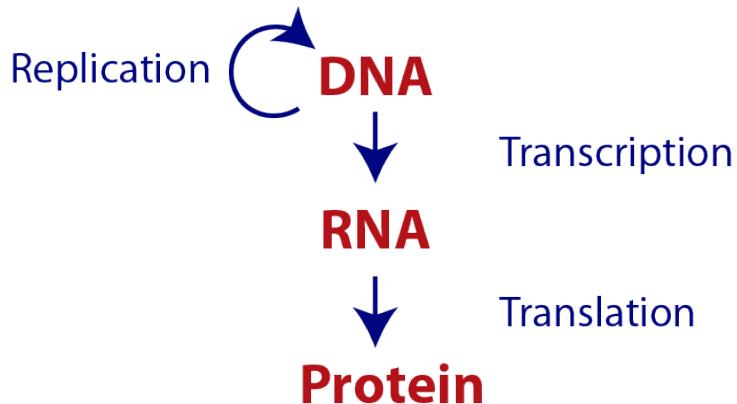
**Figure 1.2:** An animal cell and organelles (source: Kelvinsong, Wikimedia Commons)

## 5. Cellular processes

- Cell growth, cell development, cell signaling,
- Example: <http://www.nature.com/nrg/multimedia/rnai>

### Central dogma of molecular biology

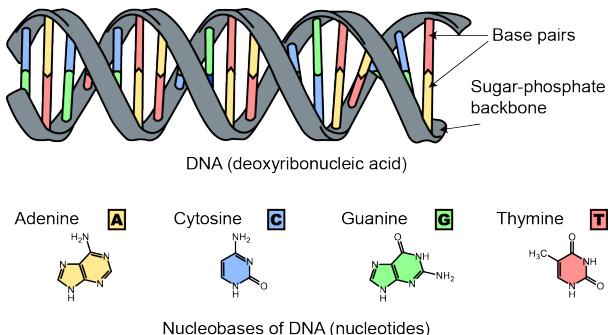
It describes the information flow within a cell.



**Figure 1.3:** Central dogma of molecular biology

### DNA (deoxyribonucleic acid)

DNA stores genetic information. It has four different bases: cytosine (C), guanine (G), adenine (A), and thymine (T).



**Figure 1.4:** DNA double helix and base pairs  
(modified from the original version by Sponk, Wikimedia Commons)

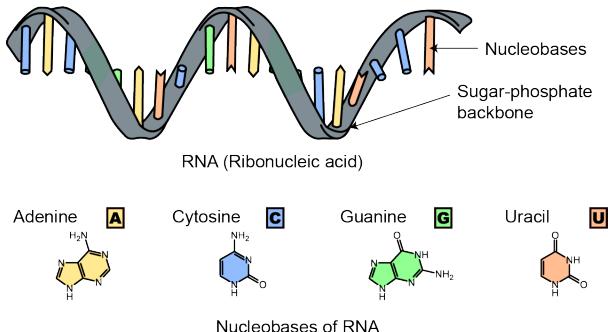
### Base pair matching (Watson-Crick base pair)

Adenine (A) pairs with thymine (T), whereas cytosine (C) pairs with guanine (G).

DNA strand1: ACGT  
||||  
DNA strand2: TGCA

## RNA (Ribonucleic acid)

RNA has various biological roles and several sub-classes. Messenger RNAs (mRNAs) convey genetic information. It has four different bases: cytosine (C), guanine (G), adenine (A), and uracil (U).



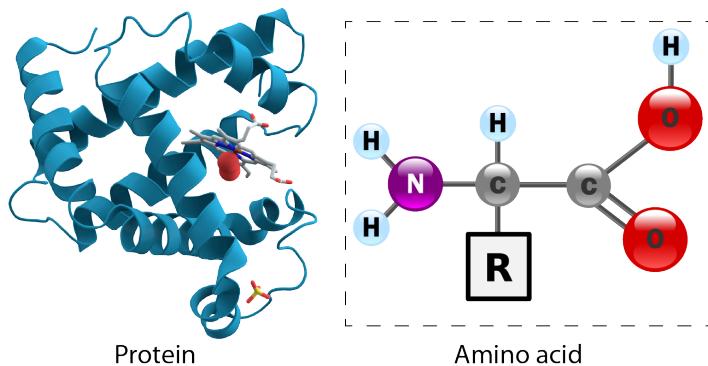
**Figure 1.5:** Single strand RNA  
(modified from the original version by Sponk, Wikimedia Commons)

## Transcription: mRNAs are transcribed from DNAs

DNA: ACGT -----> RNA: ACGU  
Transcription

## Protein

Proteins are large molecules consisting of amino acids. There are 20 common amino acids.



**Figure 1.6:** Protein 3D structure and amino acids  
(sources: AzaToth, Wikimedia Commons, YassineMrabet, Wikimedia Commons)

## Translation: Amino-acids are translated from mRNAs

mRNA: GUC -----> AA: Valine  
Translation

## Universal genetic code

A codon consists of three nucleic acids. Single-letter or three-letter names can be used for amino acids.

Gentic code				
2nd base				
	U	C	A	
3rd base in each row	U	UCU (Ser/S) Serine	UAU (Tyr/Y) Tyrosine	UGU (Cys/C) Cysteine
	UUC (Phe/F) Phenylalanine	UCC (Ser/S) Serine	UAC (Tyr/Y) Tyrosine	UGC (Cys/C) Cysteine
	UUA (Leu/L) Leucine	UCA (Ser/S) Serine	UAA Ochre (Stop)	UGA Opal (Stop)
	UUG (Leu/L) Leucine	UCG (Ser/S) Serine	UAG Amber (Stop)	UGG (Trp/W) Tryptophan
1st base	C	CUU (Leu/L) Leucine	CCU (Pro/P) Proline	CAU (His/H) Histidine
	CUC (Leu/L) Leucine	CCC (Pro/P) Proline	CAC (His/H) Histidine	CGU (Arg/R) Arginine
	CUA (Leu/L) Leucine	CCA (Pro/P) Proline	CAA (Gln/Q) Glutamine	CGC (Arg/R) Arginine
	CUG (Leu/L) Leucine	CCG (Pro/P) Proline	CAG (Gln/Q) Glutamine	CGA (Arg/R) Arginine
A	AUU (Ile/I) Isoleucine	ACU (Thr/T) Threonine	AAU (Asn/N) Asparagine	AGU (Ser/S) Serine
	AUC (Ile/I) Isoleucine	ACC (Thr/T) Threonine	AAC (Asn/N) Asparagine	AGC (Ser/S) Serine
	AUA (Ile/I) Isoleucine	ACA (Thr/T) Threonine	AAA (Lys/K) Lysine	AGA (Arg/R) Arginine
	AUG (Met/M) Methionine	ACG (Thr/T) Threonine	AAG (Lys/K) Lysine	AGG (Arg/R) Arginine
G	GUU (Val/V) Valine	GCU (Ala/A) Alanine	GAU (Asp/D) Aspartic acid	GGU (Gly/G) Glycine
	GUC (Val/V) Valine	GCC (Ala/A) Alanine	GAC (Asp/D) Aspartic acid	GGC (Gly/G) Glycine
	GUA (Val/V) Valine	GCA (Ala/A) Alanine	GAA (Glu/E) Glutamic acid	GGA (Gly/G) Glycine
	GUG (Val/V) Valine	GCG (Ala/A) Alanine	GAG (Glu/E) Glutamic acid	GGG (Gly/G) Glycine

**Figure 1.7:** Universal genetic code  
(modified from the original version by Häggström, Wikimedia Commons)

## Cellular functions of proteins

- Enzymes: catalyze chemical reaction
- Cell signaling: hormone (e.g. insulin), antibodies,
- Structural: collagen, cartilage, keratin,

## Exercises 1.1

1. Draw a simple diagram of the central dogma of molecular biology and briefly explain the information flow of the molecules.

2. What are the DNA sequences of the opposite strand for the following DNA sequences?

Seq1 CCGATT  
Seq2 TTACGC  
Seq3 ACGCGC

3. What are the mRNA sequences transcribed from the following DNA sequences?

4. What are the polypeptide sequences translated from the following mRNA sequences?  
Answer them with both one-letter and three letter names.

Seq1 AUGUUUUAA  
Seq2 GCAGCAAAAA

## 1.2 Introduction to Biotechnology

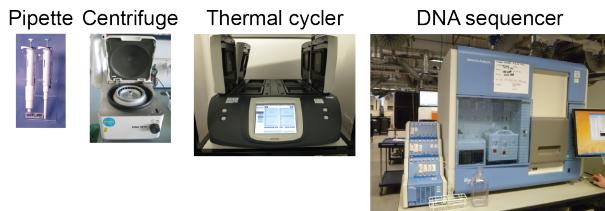
Biotechnology is the use of laboratory techniques to study living organism and cells.

### Applications of biotechnology

Branches of biotechnology can be explained with different colors.

- Red: medical processes
- Green: agricultural processes
- White: industrial processes
- Blue: marine and aquatic applications

### Laboratory tools and equipment



**Figure 1.8:** Pipette, centrifuge, thermal cycler, and DNA sequencer  
(sources: Domain, Manske, Rrror, RE73 via Wikimedia Commons)

### Human genome project

It was a large-scale international research project to determine the whole DNA sequences of human.

- 1990 - 2003
- \$2.7 billion

### Next generation sequencing

Sequence technologies have been rapidly advanced since the human genome project.

Example: sequence a whole human genome with Illumina HiSeq X Ten.

- One day
- \$1000

## Protein sequencing

Proteins are generally more studied than DNAs and RNAs, but the whole proteome is generally harder to analyze than the whole genome. MS (mass-spectrometry) based technologies are widely used to sequence proteins.



**Figure 1.9:** Orbitrap mass spectrometer (source: Wiòrkiewicz, Wikimedia Commons)

## 1.3 Bioinformatics

Bioinformatics uses computational approaches to solve problems in life sciences. It is based on computer science.

### Similar or almost equivalent disciplines

- Biostatistics
- Biophysics
- Systems biology
- Computational biology

### Not much related with bioinformatics

- Health informatics
- Forensic science

### Scope of INF281

We mainly cover the following fields of bioinformatics in this course.

- Pairwise alignment
- Database search
- Statistical evaluation
- Multiple alignment
- Phylogenetic tree
- Scoring scheme
- Sequence patterns

## Popular bioinformatics programs

BLAST and ClustalW are popular tools for sequence analysis.

- BLAST: a program for database search  
URL: <http://blast.ncbi.nlm.nih.gov>
- ClustalW: a program for multiple alignments  
URL: <http://www.ch.embnet.org/software/ClustalW.html>

Rank	Title	Times cited
1	Protein measurement with the folin phenol reagent	305148
2	Cleavage of structural proteins during the assembly of the head of bacteriophage T4	213005
3	A rapid and sensitive method for the quantitation of microgram quantities of protein utilizing the principle of protein-dye binding	155530
4	DNA sequencing with chain-terminating inhibitors	65335
5	Single-step method of RNA isolation by acid guanidinium thiocyanate-phenol-chloroform extraction	60397
6	Electrophoretic transfer of proteins from polyacrylamide gels to nitrocellulose sheets: procedure and some applications	53349
7	Development of the Colle-Salvetti correlation-energy formula into a functional of the electron density	46702
8	Density-functional thermochemistry. III. The role of exact exchange	46145
9	A simple method for the isolation and purification of total lipides from animal tissues	45131
10	<b>Clustal W:</b> improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice	40289
11	Nonparametric estimation from incomplete observations	38600
12	<b>Basic local alignment search tool</b>	38380
13	A short history of SHELX	37978
14	<b>Gapped BLAST and PSI-BLAST:</b> A new generation of protein database search programs	36410
15	A revised medium for rapid growth and bio assays with tobacco tissue cultures	36132

**Table 1.1:** The 15 most cited papers of all time  
(The top 100 papers, Van Noorden, Maher, and Nuzzo, *Nature*, 2014)

## Part II

# 2 Global pairwise alignment

## 2.1 Pairwise alignment

A pairwise alignment is a basic sequence structure that consists of two sequences. A global alignment stretches to the whole part of two sequences, whereas a local alignment usually contains only part of the sequences.

### Components of pairwise alignment

We name two sequences as database or d and query or q through this course. They may represent sequences from two different species or organisms.

Identical sequences.

q: ACGT  
d: ACGT

One mismatch.

q: ACGT  
d: ACGA

The '-' symbol represents a blank. A single or a set of multiple blanks further represents a gap, which is an indication of insertion or deletion in the course of evolution between two organisms.

q: ACGT  
d: A-GT

**N.B.** A gap cannot be aligned with another gap.

### Example of a simple scoring scheme

- Match: 1
- Mismatch: 0
- Gap penalty: 1 (use -1 for the actual calculation)

We may use the following notation.

- $R_{ab} = 1$  for  $a = b$
- $R_{ab} = 0$  for  $a \neq b$
- $g = 1$

### Exercise 2.1

Use the simple scoring scheme above and calculate the scores of the following two alignments.

Alignment 1

q: GCA-GCA  
d: GA-TG-A

Alignment 2

q: GCA-GCA  
d: G-ATG-A

## 2.2 Alignment by brute-force

A brute-force approach finds the alignment with the highest score by simply considering all possible alignments and calculates the score for each of them.

### An example of brute-force approach

We find the optimal alignment for the following sequences by using the scoring scheme below.

Sequences:

q: AG, d: ACG

Scoring scheme:

$R_{ab} = 1$  for  $a = b$   
 $R_{ab} = 0$  for  $a \neq b$   
 $g = 1$

#### 1. The length of alignment

- Maximum length:  $\text{length}(q) + \text{length}(d)$
- Minimum length:  $\max(\text{length}(q), \text{length}(d))$

#### 2. All possible alignments when length = 5

---AG	A---G	A--G-	AG---	--A-G
ACG--	-ACG-	-AC-G	--ACG	AC-G-
--AG-	-AG--	-A--G	-A-G-	A-G--
AC--G	A--CG	A-CG-	A-C-G	-A-CG

#### 3. All possible alignments when length = 4

A--G	A-G-	AG--	A--G	-A-G	-AG-
ACG-	AC-G	A-CG	-ACG	ACG-	AC-G
-AG-	A-G-	--AG	--AG	-A-G	AG--
A-CG	-ACG	ACG-	AC-G	A-CG	-ACG

#### 4. All possible alignments when length = 3

-AG	A-G	AG-
ACG	ACG	ACG

#### 5. Alignment with the best score

ACG  
A-G

Score: 1

### Search space size of the brute-force approach

The search space size is the number of all possible alignments. It is 25 ( $10 + 12 + 3$ ) for the example above.

### Rapid growth of search space size

#### Example 1

q: ACGACG, d: AGAG

Search space size: 1289

#### Example 2

q: ACGACGACGACG, d: AGAGAGAG

Search space size: 4,673,345

### Exercise 2.2

Find the alignment with the best score for the sequences. Use the simple scoring scheme below.

Sequences:

q: A, d: AC

Scoring scheme:

$$R_{ab} = 1 \text{ for } a = b$$

$$R_{ab} = 0 \text{ for } a \neq b$$

$$g = 1$$

1. What are the maximum and minimum lengths of the alignment?
2. Identify all possible alignments.
3. What is the best score?
4. What is the search space size when the brute-force approach is used?

## 2.3 Table representation of alignment

Several data structures can be used to represent an alignment. The table representation is frequently used and also makes the process clear when we combine it with dynamic programming (DP) later.

### Data structures and algorithms

It is important to consider the following aspects before solving computational problems.

1. Identify and analyze the problem you want to solve
2. Pick up an algorithm that can efficiently solve the problem
3. Decide a data structure that works with the algorithm of your choice

We use a table format (2D array) to solve global alignments by dynamic programming.

## Example of table format

Alignment:

q: -AG-  
d: A-CG

### 1. Initial setup

1. Make a table with the size of  $(1 + \text{length}(q))$  by  $(1 + \text{length}(d))$
2. Add the database sequence as column labels
3. Add the query sequence as row labels

q/d		A	C	G
A	S			
G				E

### 2. Add arrows

We use three types of arrows to form an alignment.

- Move diagonally: add the letters from q and d to the alignment
- Move vertically: add - and the letter from d to the alignment
- Move horizontally: add the letter from q and - to the alignment

It should start from S and stop at E.

q/d		A	C	G
A	S →			
		↓	↘	→ E
G				

### Exercise 2.3

Find the corresponding alignments for Table 1, 2 and 3.

Table 1

q/d		A	C	G
A	S ↘			
		↖	→ E	
G				

Table 2

q/d		A	C	G
A	S →	→	→	↓
				↓
G				

Table 3

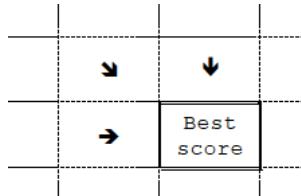
q/d		A	C	G
A	S ↓			
		→	→	→ E
G				

## 2.4 Global alignment with DP

Dynamic programming (DP) provides a solution for a multi-stage decision process, in which larger decisions recursively nest smaller decisions.

### Memorize the best score in a table cell

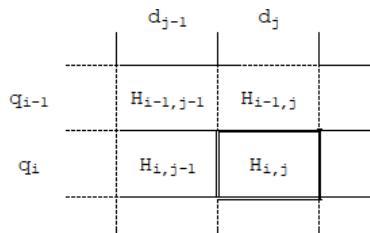
For global alignment, the core procedure of DP is updating a cell with the highest score from the three different scores calculated from its adjacent cells. DP ends when the entire table is updated.



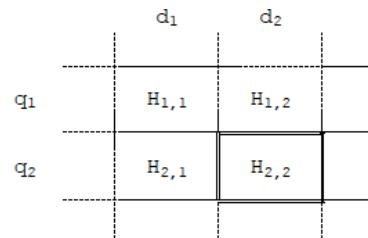
### Table notation and indices

$H_{i,j}$  represents the score of the cell for the current update.  $H_{i-1,j}$ ,  $H_{i,j-1}$ , and  $H_{i-1,j-1}$  are the scores of the adjacent cells.

Cell  $H_{i,j}$  and its adjacent cells



Example



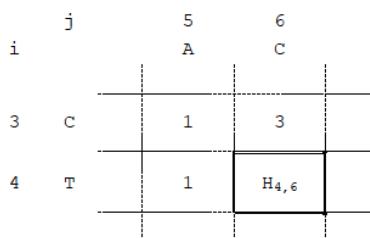
### Calculation of three candidate scores

$H_{i,j}^{(0)}$ ,  $H_{i,j}^{(1)}$ , and  $H_{i,j}^{(2)}$  represent the three candidate scores of  $H_{i,j}$ . They are respectively calculated as:

$$\begin{aligned} H_{i,j}^{(0)} &= H_{i-1,j} - g && \text{(vertical)} \\ H_{i,j}^{(1)} &= H_{i,j-1} - g && \text{(horizontal)} \\ H_{i,j}^{(2)} &= H_{i-1,j-1} + R_{a,b} && \text{(diagonal)} \end{aligned}$$

### Exercise 2.4

Calculate the scores of  $H_{4,6}^{(0)}$ ,  $H_{4,6}^{(1)}$ , and  $H_{4,6}^{(2)}$  first and then update  $H_{4,6}$ .



Scoring scheme:

$$R_{ab} = 1 \text{ for } a = b$$

$$R_{ab} = 0 \text{ for } a \neq b$$

$$g = 1$$

## Initialization

The first row and the first column can be calculated independently from the adjacent cells.

$$H_{0,j} : j * -1 * g$$

$$H_{i,0} : i * -1 * g$$

Example

		j	0	1	2
		i	A	C	
		0	0	-1	-2
0	G	-1			
1	T	-2			

## Exercise 2.5

Update all cells of Table 1 and 2. Use the scoring scheme in Exercise 2.4.

Table 1

		A	C
G			

Table 2

		A
G		
T		

## Sub-solutions

In DP, larger decisions recursively nest smaller decisions. For instance, Table S is included in Table L.

Table S

		A
A	$H_{0,0}$	$H_{0,1}$
	$H_{1,0}$	$H_{1,1}$

Table L

		A	G
A	$H_{0,0}$	$H_{0,1}$	$H_{0,2}$
	$H_{1,0}$	$H_{1,1}$	$H_{1,2}$
C	$H_{2,0}$	$H_{2,1}$	$H_{2,2}$

## Pseudo-code of updating DP table for global alignment

---

### Algorithm 2.1: Update dynamic programming table for global alignment

---

$H_{i,j}$  : Dynamic programming table  
 $R_{a,b}$ : Match/mismatch scores  
 $g$  : Gap penalty

```
// Initialization
for i ← 0 to m do
    |  $H_{i,0} \leftarrow i * -1 * g;$ 
end
for j ← 1 to n do
    |  $H_{0,j} \leftarrow j * -1 * g;$ 
end

// Main loop for table update
for i ← 1 to m do
    for j ← 1 to n do
        |  $H_{i,j} \leftarrow \max(H_{i-1,j} - g, H_{i,j-1} - g, H_{i-1,j-1} + R_{a,b});$ 
    end
end
```

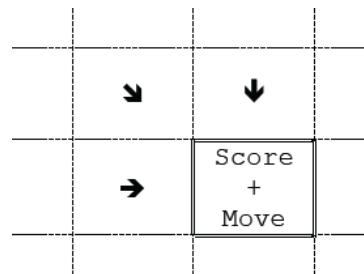
---

## 2.5 Backtracking

Backtracking is a post-processing procedure to find the alignments that have yielded the best score.

### Store movement in cells

A table cell can be used for storing the movement.

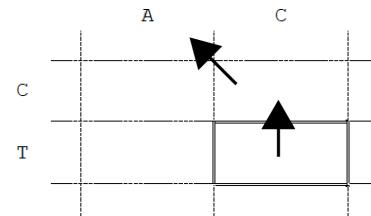


### Example

Cells with scores and directions

	A	C
C	Score:1 Move:V	Score:3 Move:D
T	Score:0 Move:V	Score:2 Move:V

Use arrows to indicate backtracking



## Exercise 2.6

Complete the DP table with scores and directions. What is the alignment with the best score?

	A	C
A		

Scoring scheme:

$$R_{ab} = 1 \text{ for } a = b$$

$$R_{ab} = 0 \text{ for } a \neq b$$

$$g = 1$$

## Re-calculate candidate scores

Re-calculating the three candidate scores also reveals the movement.

$$H_{i,j}^{(0)} = H_{i-1,j} - g \quad (\text{vertical})$$

$$H_{i,j}^{(1)} = H_{i,j-1} - g \quad (\text{horizontal})$$

$$H_{i,j}^{(2)} = H_{i-1,j-1} + R_{a,b} \quad (\text{diagonal})$$

## Example

	A	C
C	1	3
T	1	2

$$H_{i,j}^{(0)} = 3 - 1 = 2 = H_{i,j} \quad \checkmark \text{ (vertical)}$$

$$H_{i,j}^{(1)} = 1 - 1 = 0 \neq H_{i,j} \quad \text{(horizontal)}$$

$$H_{i,j}^{(2)} = 1 + 0 = 1 \neq H_{i,j} \quad \text{(diagonal)}$$

## Common mistake with backtracking

For the re-calculation approach, it is not to find  $\max(H_{i-1,j}, H_{i,j-1}, H_{i-1,j-1})$ . You must re-calculate the candidates and then  $\max(H_{i,j}^{(0)}, H_{i,j}^{(1)}, H_{i,j}^{(2)})$  to find the actual direction.

## Implementation with recursive call

Recursive calls are usually used to implement DP backtracking.

---

### Algorithm 2.2: DP backtracking

---

$S_q$  : Sequence q  
 $S_d$  : Sequence d  
 $H_{i,j}$  : Dynamic programming table  
 $R_{a,b}$ : Match/mismatch scores  
 $g$  : Gap penalty

```

proc backTrack(i, j,  $A_q$ ,  $A_d$ , k)
  i : Index of sequence q
  j : Index of sequence d
   $A_q$  : q part of alignment (stored in reverse order)
   $A_d$  : d part of alignment (stored in reverse order)
  k : Index for  $A_q$  and  $A_d$ 

  //
  // Need to implement recursion termination here
  // ...
  //

  if  $H_{i,j} = H_{i-1,j} - g$  then                                // vertical
     $A_{q,k} \leftarrow S_{q,i};$ 
     $A_{d,k} \leftarrow '-';$ 
    backTrack(i - 1, j,  $A_q$ ,  $A_d$ , k + 1);
  end

  if  $H_{i,j} = H_{i,j-1} - g$  then                                // horizontal
     $A_{q,k} \leftarrow '-';$ 
     $A_{d,k} \leftarrow S_{d,i};$ 
    backTrack(i, j - 1,  $A_q$ ,  $A_d$ , k + 1);
  end

  if  $H_{i,j} = H_{i-1,j-1} + R_{S_{q,i},S_{d,i}}$  then          // diagonal
     $A_{q,k} \leftarrow S_{q,i};$ 
     $A_{d,k} \leftarrow S_{d,i};$ 
    backTrack(i - 1, j - 1,  $A_q$ ,  $A_d$ , k + 1);
  end
end
    
```

---

## Exercise 2.7

Find the alignment with the best score.

	A	C
G		
T		

Scoring scheme:

$$\begin{aligned}
 R_{ab} &= 1 \text{ for } a = b \\
 R_{ab} &= 0 \text{ for } a \neq b \\
 g &= 1
 \end{aligned}$$

## 2.6 Needleman-Wunsch algorithm

The method of using DP to solve global pairwise alignment is called the Needleman-Wunsch algorithm in the field of bioinformatics.

### Complexity

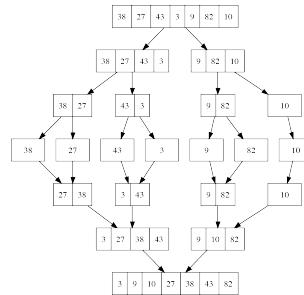
- Time:  $O(nm)$
- Space:  $O(nm)$

### Comparisons with other algorithms

The Needleman-Wunsch algorithm is similar to several algorithms.

### Divide and conquer algorithms

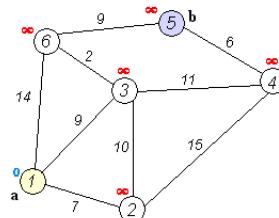
Sub-solutions must be independent with divide and conquer.



**Figure 2.1:** Merge sort (source: VineetKumar, Wikimedia Commons)

### Dijkstra's algorithm

Worst-case performance of Dijkstra:  $O(|E| + |V| \log |V|)$



**Figure 2.2:** Dijkstra's algorithm (source: Ibmua, Wikimedia Commons)

## 3 Extension of global alignment

### 3.1 Introduction of score matrix

We will expand our simple scoring scheme to score matrices. This expansion allows us to solve general alignment problems with DNA, RNA, and protein sequences.

#### Extension of a scoring scheme to a score matrix

The matrix below is equivalent with match: 1 and mismatch: 0.

	a	b
a	1	0
b	0	1

#### Example of a DNA score matrix

The matrix below is equivalent with match: 5 and mismatch: -4.

	A	T	G	C
A	5	-4	-4	-4
T		5	-4	-4
G			5	-4
C				5

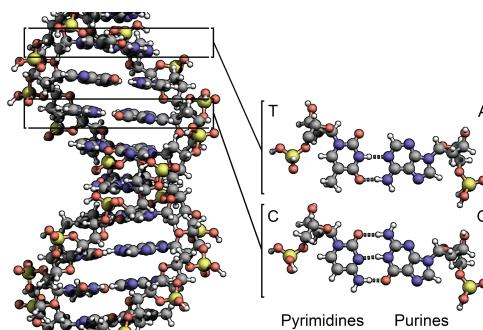
#### Applications of score matrix

Score matrices are more flexible than the simple scoring scheme. For instance, they can be used for the following cases.

- DNA pairs
- RNA pairs
- Similarity of protein sequences by amino acid properties

#### DNA pairs (Watson-Crick pairs)

A thymine pairs with an adenine, and a cytosine pairs with a guanine.



**Figure 3.1:** Watson-Crick pairs (source: Zephyris, Wikimedia Commons)

### Example of score matrix for DNA pairs

The matrix reflects the difference of hydrogen bonds.

	A	T	G	C
A	5	-4	-4	-4
T		5	-4	-4
G			5	-4
C				5

### Example of DP for DNA pairs

You can use DP to find a DNA alignment with Watson-Crick pairs. For instance, the DP table below is used to solve the optimal alignment for two DNA sequences:  $q = AC$  and  $d = GT$  with gap penalty  $g = 4$ .

DP table:

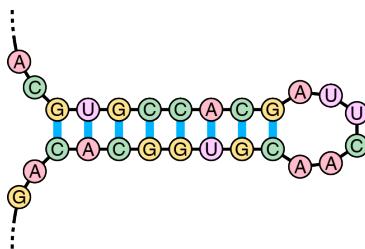
q/d		G	T
		0	-4
A	-4	-3	0
	C	-8	1

Alignment:

q: AC-  
d: -GT

### RNA pairs

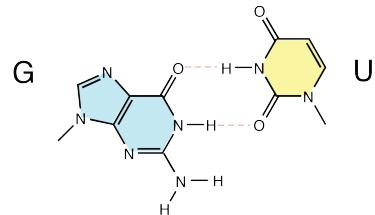
A single stand of RNA can form a 3D structure that has a biological function. The secondary structure of RNA is a two-dimensional representation of the structure.



**Figure 3.2:** RNA stem-loop (source: Sakurambo, Wikimedia Commons)

### Wobble pairs

Wobble pairs are not canonical Watson-Crick pairs, but they can still form hydrogen bonds.



**Figure 3.3:** GU wobble pairs

(modified from the original version by Fdardel, Wikimedia Commons)

### Example of DP for RNA pairs

You can form the following DP table for two RNA sequences:  $q = AU$  and  $d = UGA$  with gap penalty  $g = 9$ .

DP table:

q/d		U	G	A
		0	-9	-18
A	-9	5	-4	-11
	-18	-4	7	1

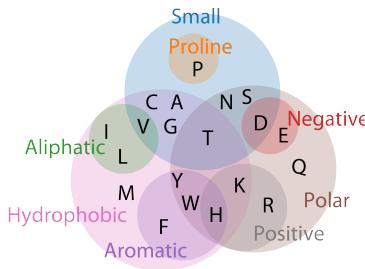
Alignment:

q: A-U

d: UGA

### Similarity of protein sequences

Amino acids can be categorized into several groups by their properties. Proteins alignments often need to take these properties into consideration.



**Figure 3.4:** Venn diagram of amino acid properties

### Example of a protein score matrix

It can be used to compare the similarity between two protein sequences.

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	13	6	9	9	5	8	9	12	6	8	6	7	7	4	11	11	11	2	4	9
R	3	17	4	3	2	5	3	2	6	3	2	9	4	1	4	4	3	7	2	2
N	4	4	6	7	2	5	6	4	6	3	2	5	3	2	4	5	4	2	3	3
D	5	4	8	11	1	7	10	5	6	3	2	5	3	1	4	5	5	1	2	3
C	2	1	1	1	52	1	1	2	2	2	1	1	1	1	2	3	2	1	4	2
Q	3	5	5	6	1	10	7	3	7	2	3	5	3	1	4	3	3	1	2	3
E	5	4	7	11	1	9	12	5	6	3	2	5	3	1	4	5	5	1	2	3
G	12	5	10	10	4	7	9	27	5	5	4	6	5	3	8	11	9	2	3	7
H	2	5	5	4	2	7	4	2	15	2	2	3	2	2	3	3	2	2	3	2
I	3	2	2	2	2	2	2	2	2	10	6	2	6	5	2	3	4	1	3	9
L	6	4	4	3	2	6	4	3	5	15	34	4	20	13	5	4	6	6	7	13
K	6	18	10	8	2	10	8	5	8	5	4	24	9	2	6	8	8	4	3	5
M	1	1	1	0	1	1	1	1	2	3	2	6	2	1	1	1	1	1	2	
F	2	1	2	1	1	1	1	1	3	5	6	1	4	32	1	2	2	4	20	3
P	7	5	5	4	3	5	4	5	5	3	3	4	3	2	20	6	5	1	2	4
S	9	6	8	7	7	6	7	9	6	5	4	7	5	3	9	10	9	4	4	6
T	8	5	6	6	4	5	5	6	4	6	4	6	5	3	6	8	11	2	3	6
W	0	2	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	55	1	0
Y	1	1	2	1	3	1	1	1	3	2	2	1	2	15	1	2	2	3	31	2
V	7	4	4	4	4	4	4	4	5	4	15	10	4	10	5	5	5	72	4	17

**Table 3.1:** 250 PAM Mutation Matrix (Chapter 22: A model of evolutionary change in proteins, Dayhoff and Schwartz, Atlas of Protein Sequence and Structure, 1978)

### **Exercise 3.1**

1. Use the DNA score matrix below with  $g = 10$  and find the optimal alignment for  $q = \text{TG}$  and  $d = \text{TCG}$ .
2. 250 PAM mutation matrix can not be directly used for global alignments. Explain what kind of matrix you need for calculating alignment scores.