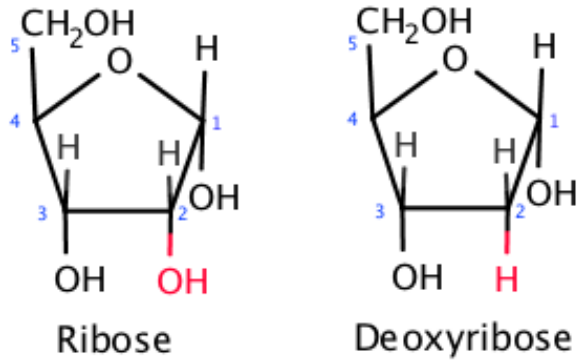
Course Overview

1. Biological Background
2. **Pairwise sequence alignment algorithms**
3. **Probabilistic alignments: Hidden Markov models**
4. **Multiple sequence alignments**
5. **Phylogeny: Algorithms for reconstructing pedigrees**
6. **Recent advances & applications:** Inferring Genetic Diversity from **Next-generation Sequencing** data.
7. **Neural nets & deep learning for sequence analysis: CNNs, RNNs, Transformers, Alphafold** etc.

Short historical Introduction

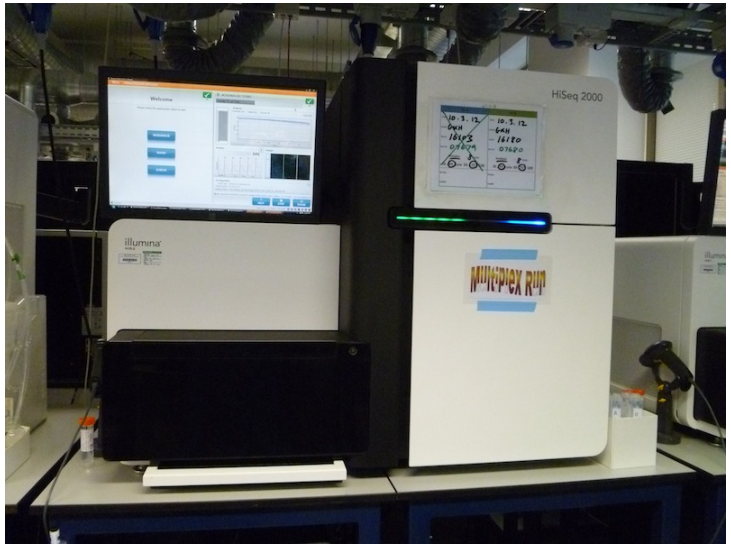
- Genetics as a natural science started in 1866: **Gregor Mendel** performed experiments that pointed to the existence of **biological elements called genes.**
- **Deoxy-ribonucleic acid (DNA)** isolated by **Friedrich Miescher** in 1869.
- 1944: Oswald Avery (and coworkers) identified DNA as the major carrier of genetic material, **responsible for inheritance.**
  Ribose: (simple) sugar molecule, deoxy-ribose $\rightsquigarrow$ loss of oxygen atom.
  Nucleic acid: overall name for DNA and RNA (large biomolecules). Named for their initial discovery in nucleus of cells, and for presence of phosphate groups (related to phosphoric acid).

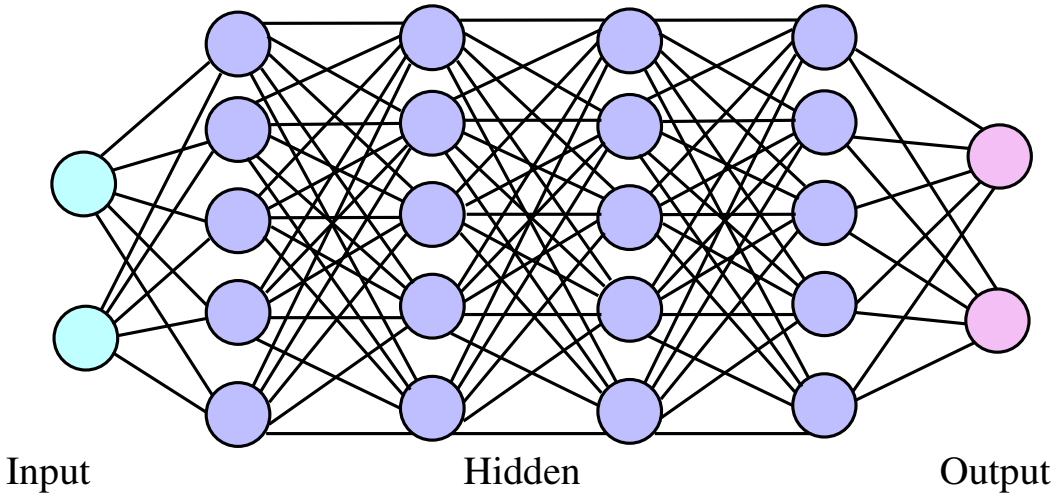


By Miranda19983 - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=84120486

Short historical Introduction

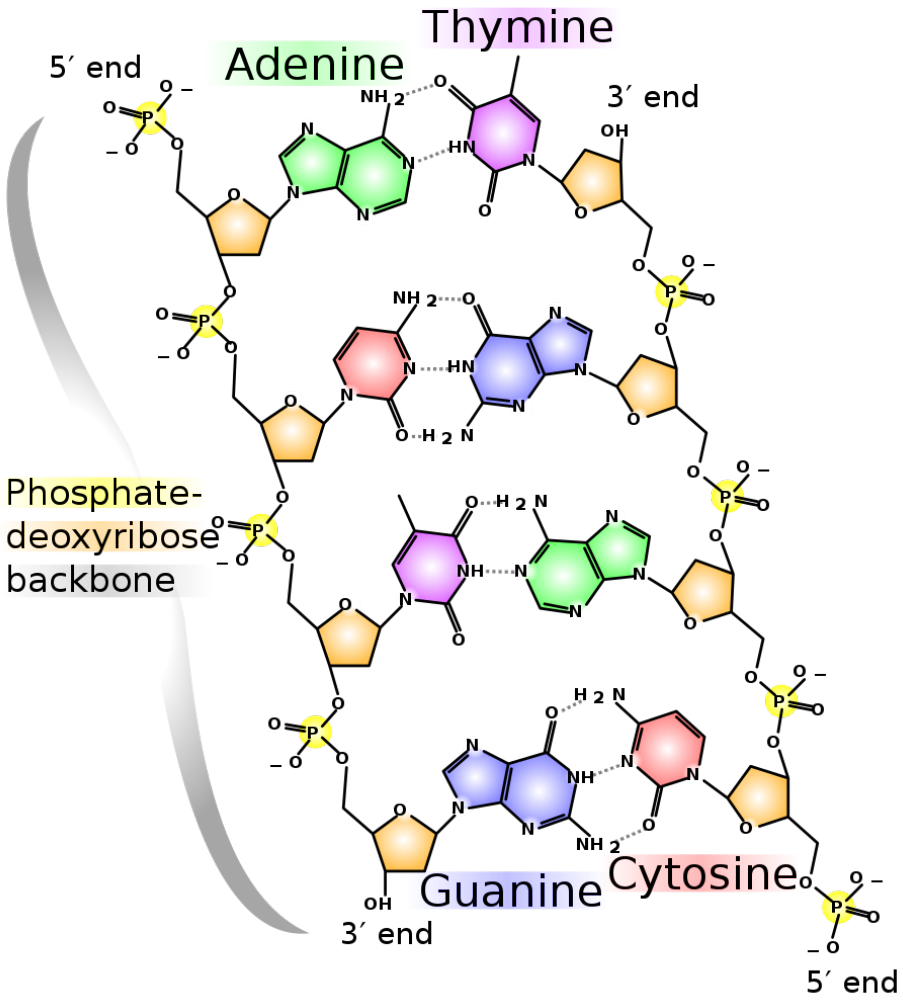
- 1953, Watson & Crick: **3-dimensional structure of DNA.** They inferred the method of **DNA replication.**
- 2001: first draft of the **human genome** published by the **Human Genome Project** and the company **Celera.**
- Many new developments, such as **Next Generation Sequencing**, **Deep learning** etc.





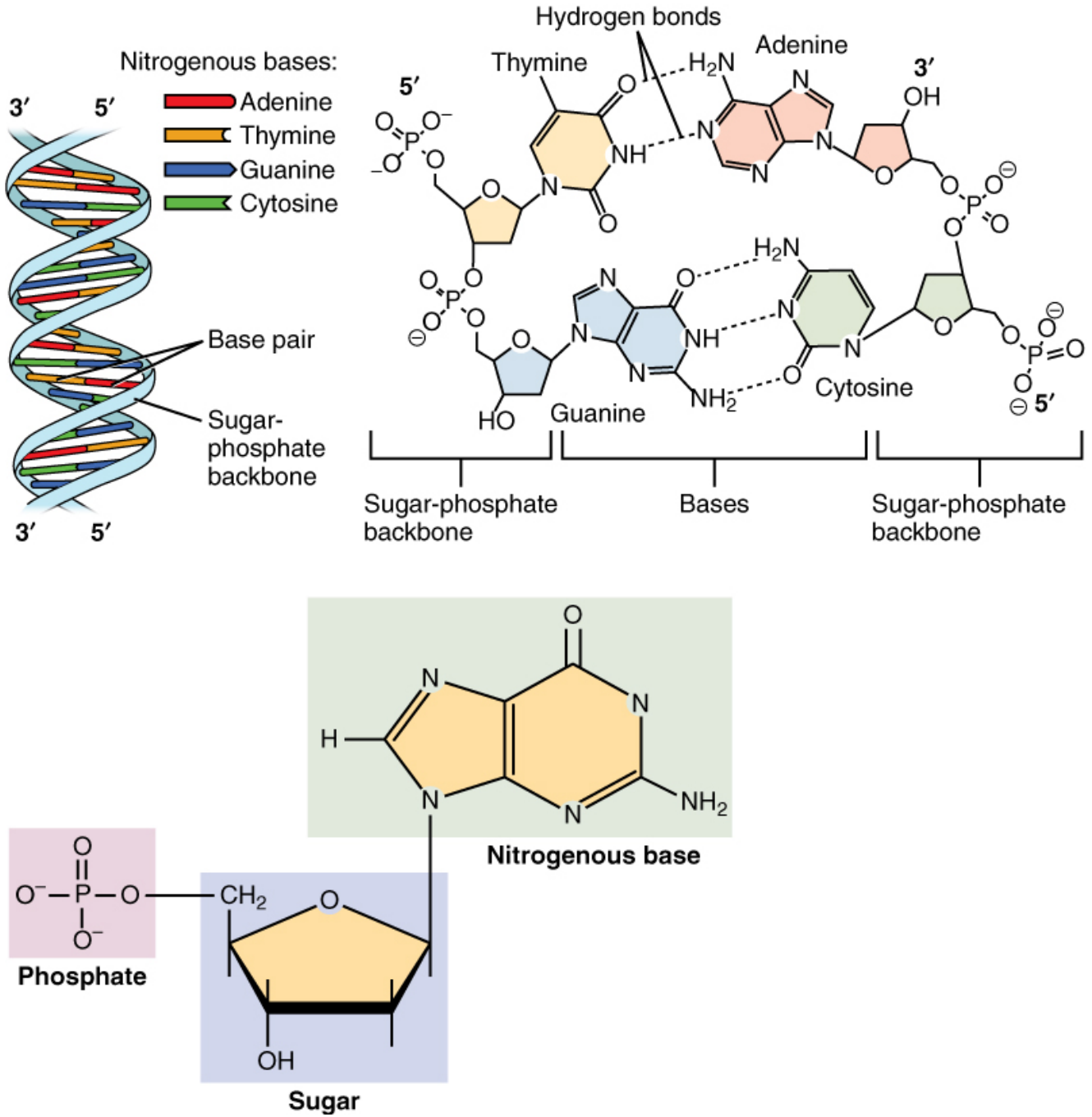
By RE73 - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=18862884

Base pairs and the DNA



By Madprime (talk · contribs) - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=1848174

- DNA composed of 4 basic molecules $\rightsquigarrow$ **nucleotides**.
- Nucleotides are identical up to different **nitrogen base:** organic molecule with a nitrogen atom that has the chemical properties of a base (due to free electron pair at nitrogen atom).
- Each nucleotide contains **phosphate**, **sugar** (of deoxy-ribose type), and one of the 4 bases: **Adenine, Guanine, Cytosine, Thymine** (A,G,C,T).
- **Hydrogen bonds** between base pairs $G \equiv C$, $A = T$.

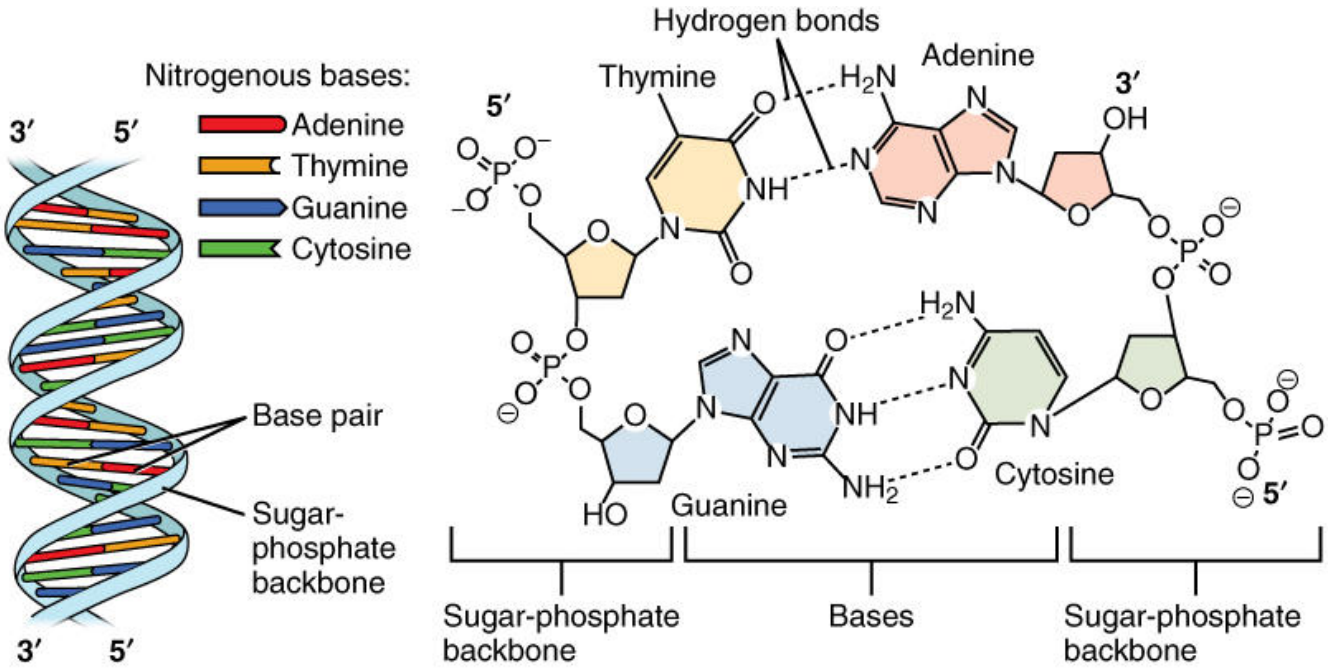


By OpenStax - https://cnx.org/contents/FPtK1zmh@8.25:fEI3C8Ot@10/Preface, CC BY 4.0, https://commons.wikimedia.org/w/index.php?curid=30131206

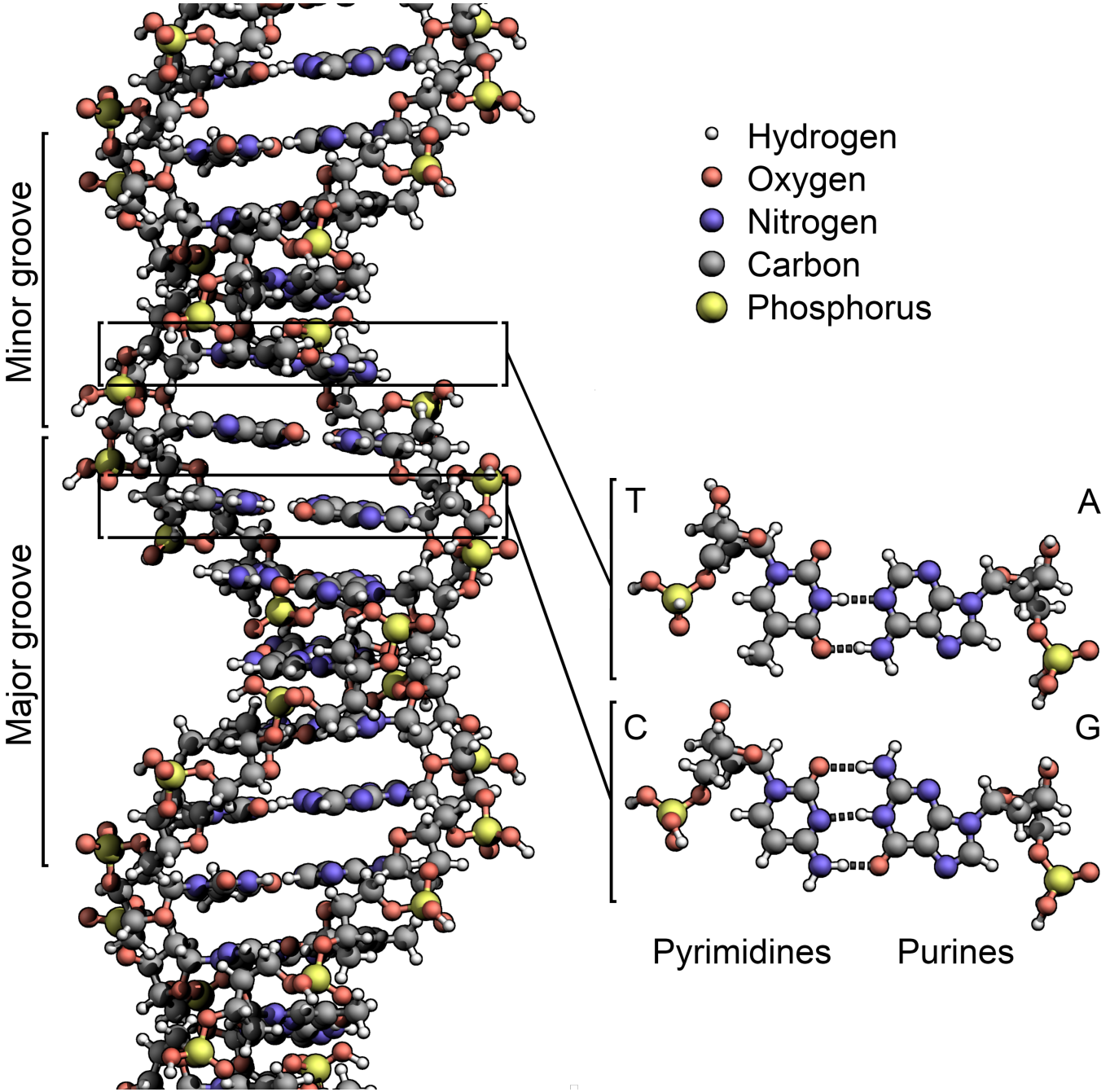
The structure of DNA

- DNA molecule is **directional** due to asymmetrical structure of the sugars which constitute the skeleton: Each sugar is connected to the strand **up-stream** in its 5th carbon and to the strand **downstream** in its 3rd carbon.

- DNA strand goes from $5'$ to $3'$. The directions of the two complementary DNA strands are reversed to one another ($\rightsquigarrow$ **Reversed Complement**).



Adapted from https://commons.wikimedia.org/w/index.php?curid=30131206



By Zephyris - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=15027555

Replication of DNA

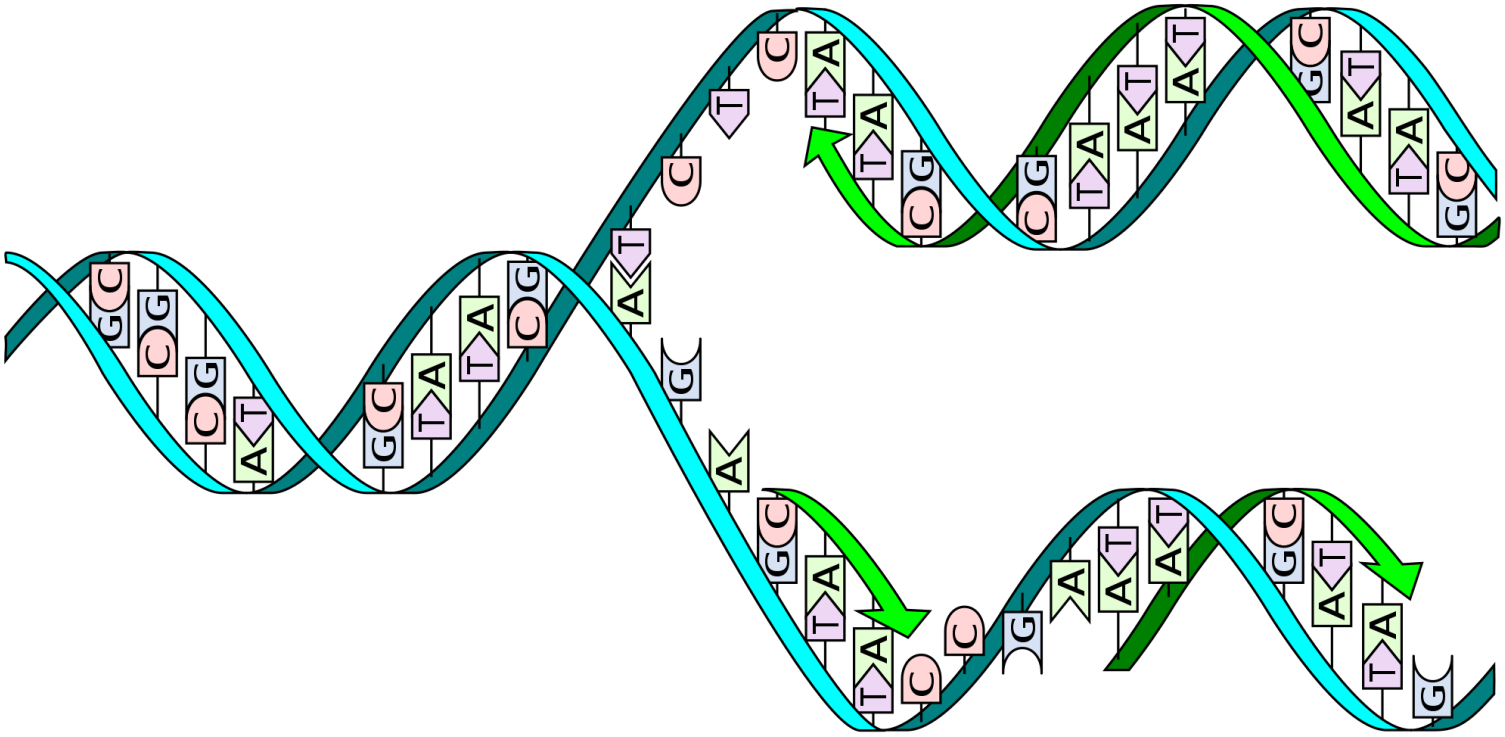
Biological process of producing two replicas of DNA from one original DNA molecule. **Cells have the distinctive property of division**

$\rightsquigarrow$ DNA replication is most essential part for **biological inheritance.**

Unwinding $\rightsquigarrow$ single bases exposed on each strand.

Pairing requirements are **strict** $\rightsquigarrow$ single strands are templates for re-forming **identical** double helix (up to **mutations**).

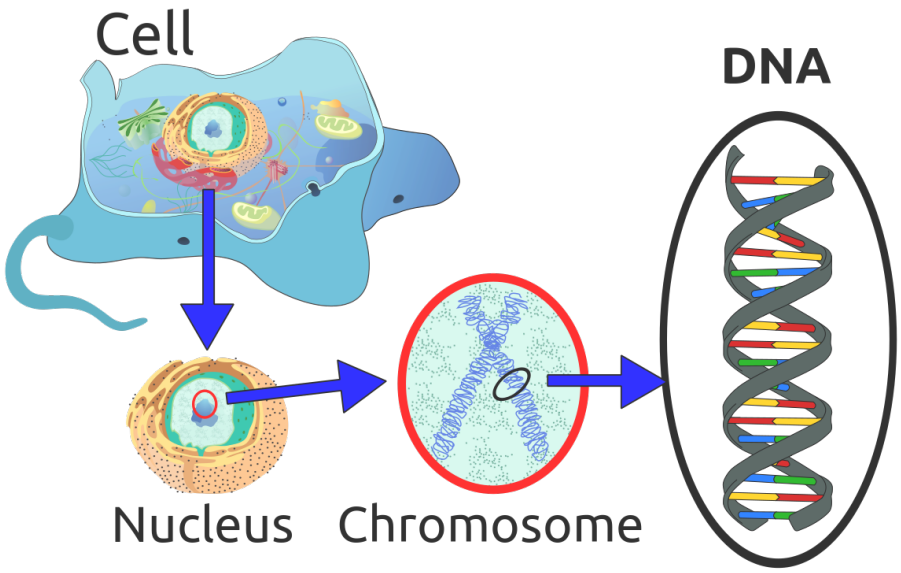
DNA polymerase: enzyme that catalyzes the synthesis of new DNA.



https://commons.wikimedia.org/w/index.php?curid=2497221

Genes and Chromosomes

- In higher organisms, DNA molecules are packed in a **chromosome**.
- **Genome:** total genetic information stored in the chromosomes.
- Every cell contains a **complete set** of the genome, differences are due to variable **expression** of genes.
- A **gene** is a sequence of nucleotides that encodes the synthesis of a gene product.

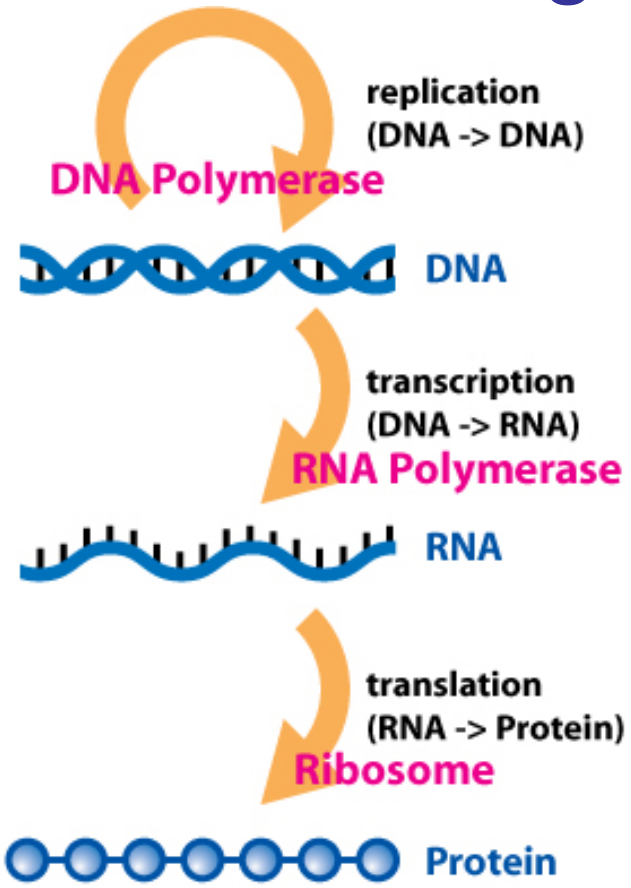


By Sponk, Tryphon, Magnus Manske,

https://commons.wikimedia.org/w/index.php?curid=20539140

- **Gene expression:** Process of synthesizing a gene product (often a protein) $\rightsquigarrow$ controls timing, location, and amount.

The Central Dogma

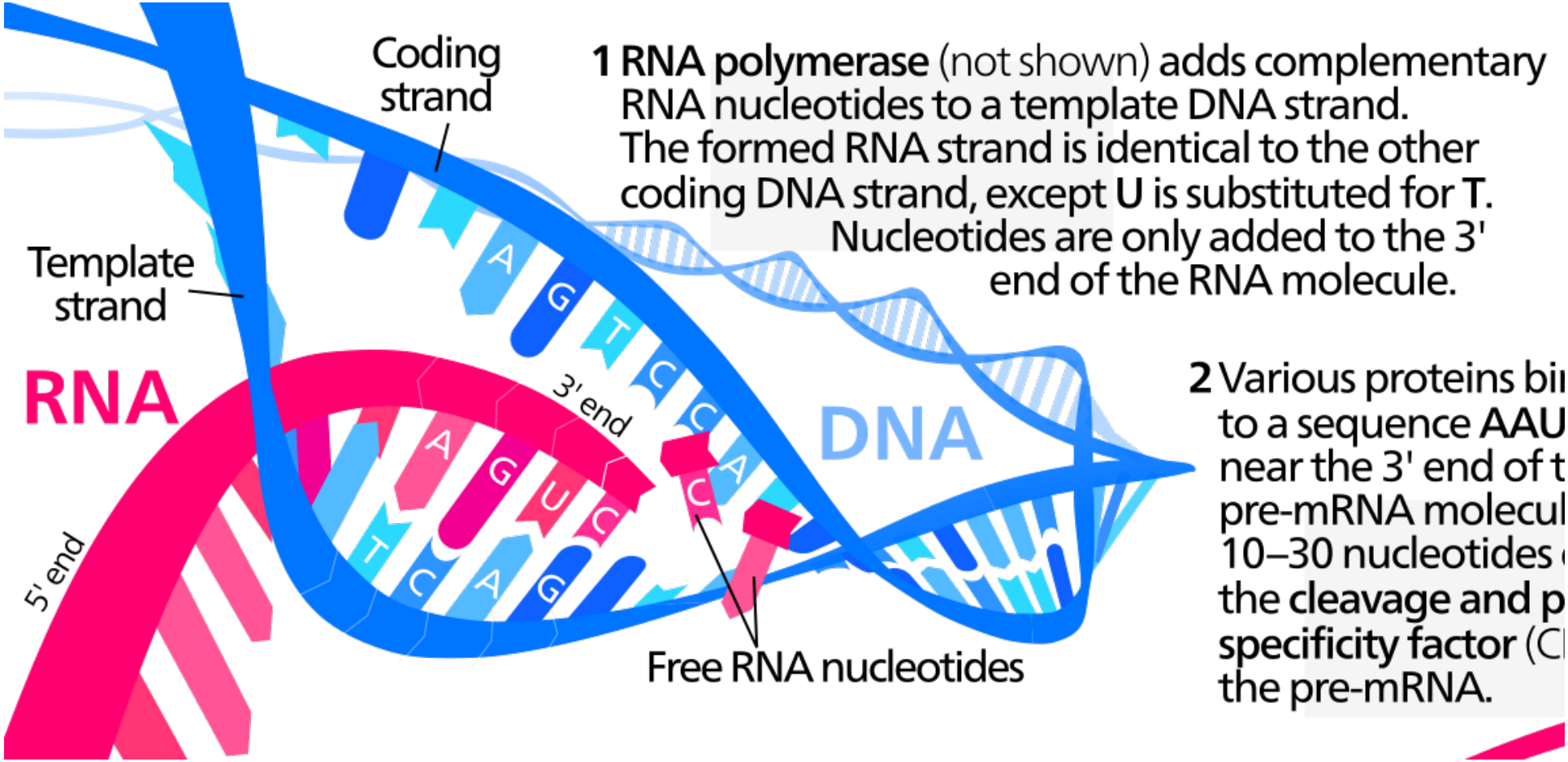


Wikipedia

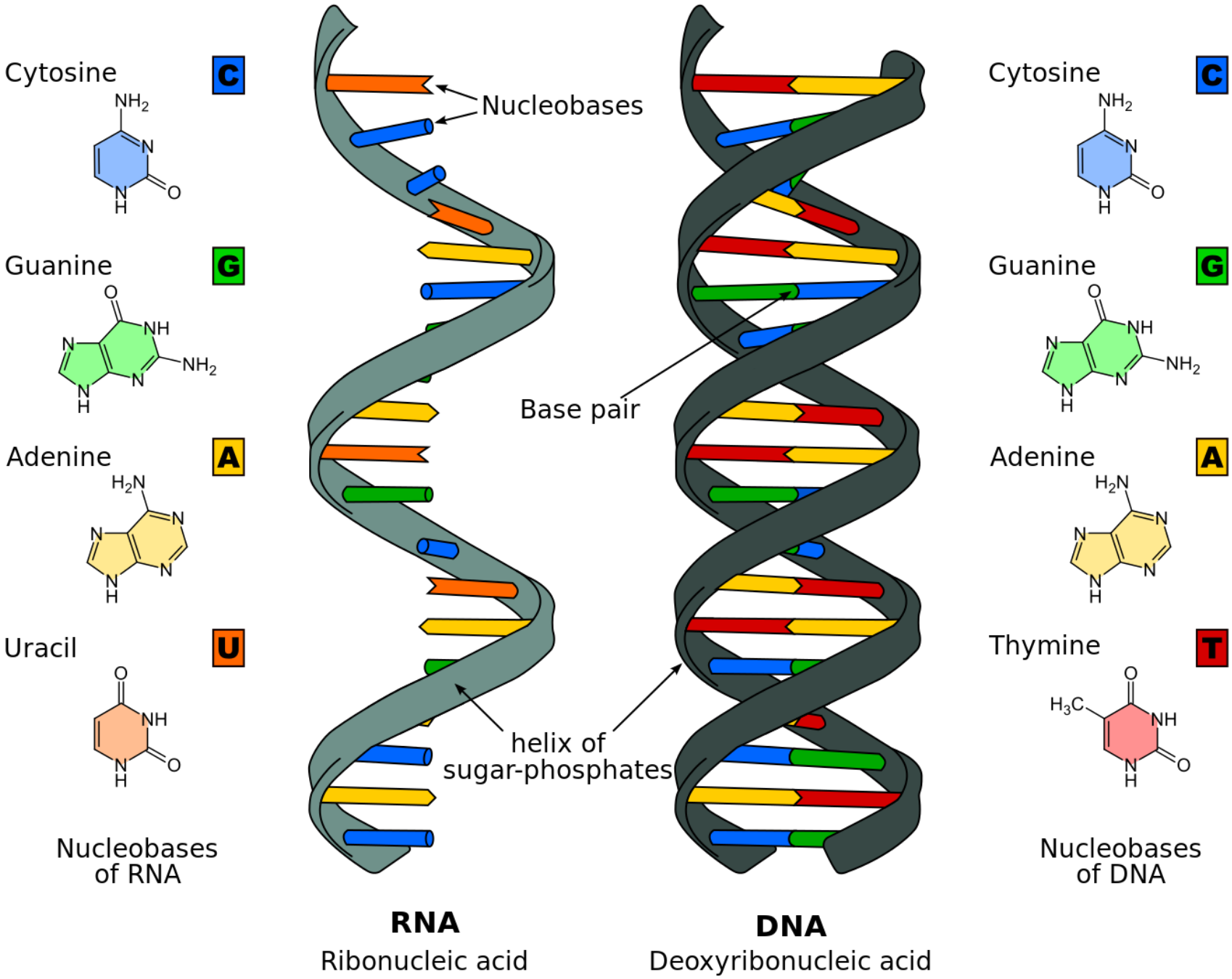
Transcription: making of an RNA molecule from DNA template.
Translation: construction of amino acid sequence from RNA.

$\Rightarrow$ Almost no exceptions ($\rightsquigarrow$ retroviruses)

Transcription



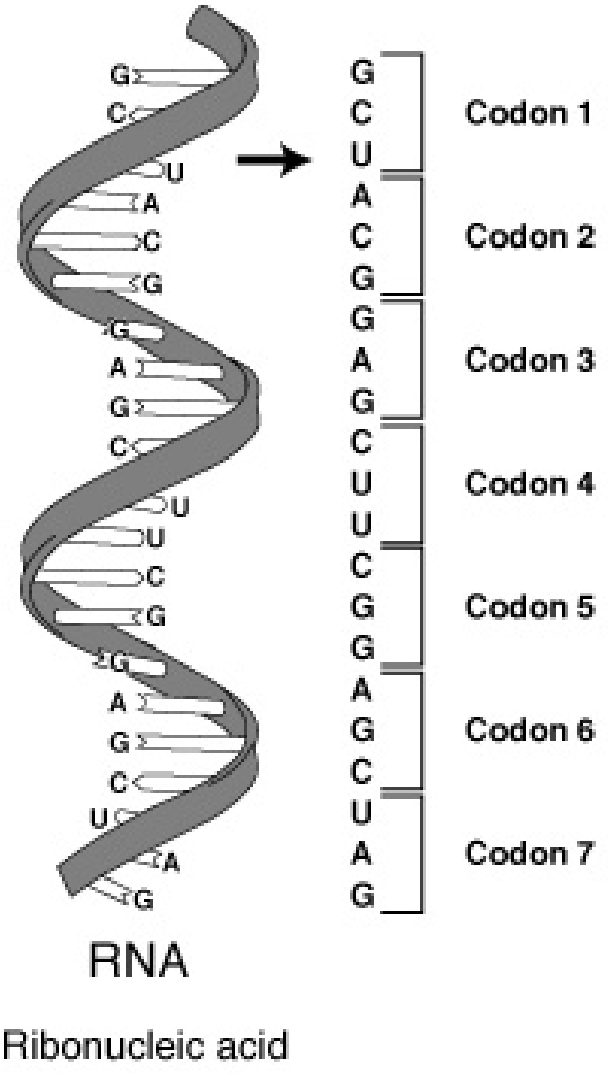
By Kelvinsong - Own work, CC BY 3.0, https://commons.wikimedia.org/w/index.php?curid=23086203



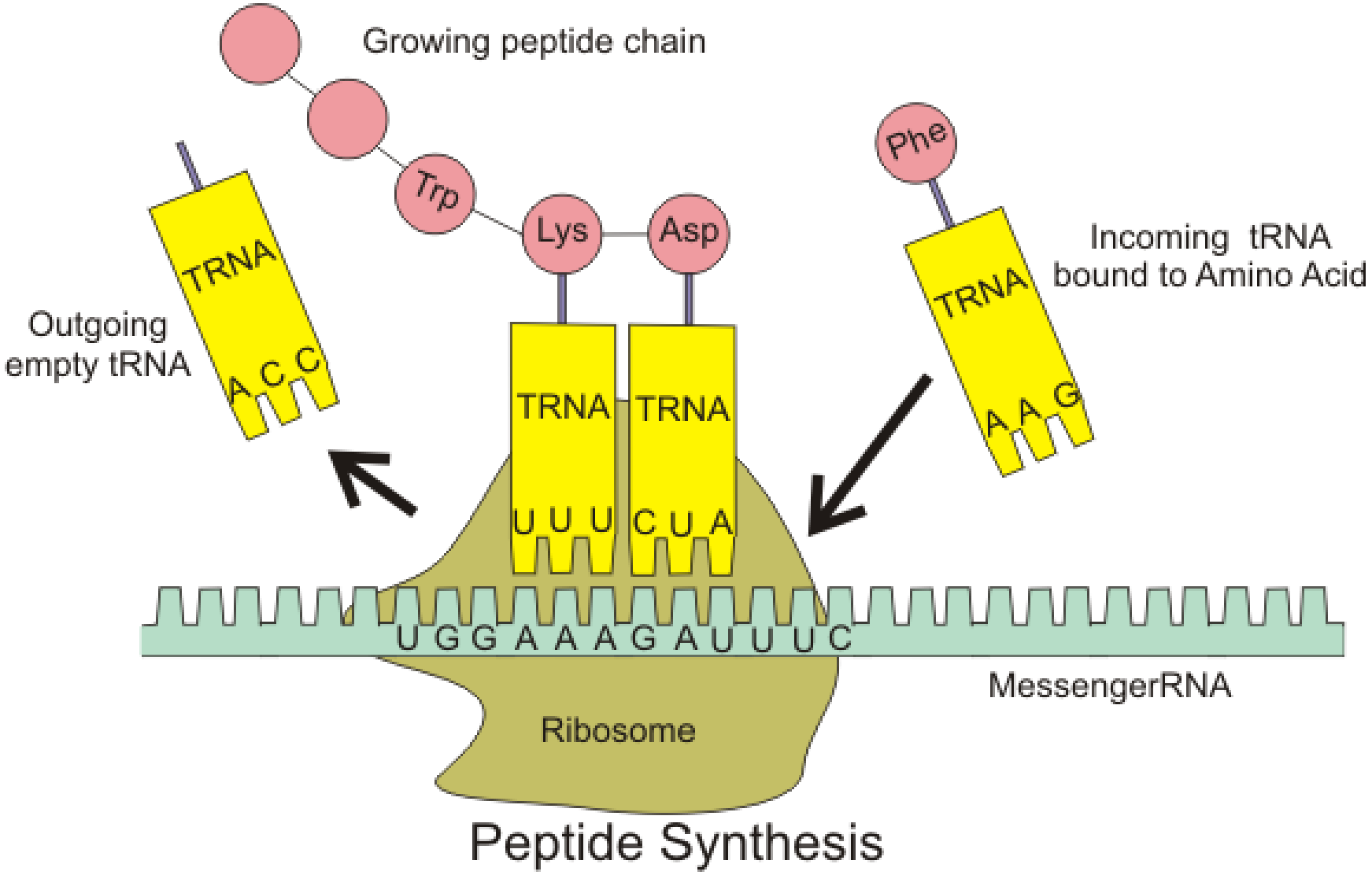
https://commons.wikimedia.org/w/index.php?curid=9810855

Translation

- mRNA molecules are translated by **ribosomes**: Enzyme that links together amino acids.
- Message is read **three bases at a time.**
- Initiated by the first AUG codon (codon = nucleotide triplet).
- Covalent bonds (=sharing of electron pairs) are made between adjacent amino acids ⇒ **growing chain of amino acids** (“polypeptide”).
- When a **“stop” codon** (UAA, UGA, UAG) is encountered, translation stops.



Wikipedia



By Boumphreyfr - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=7200200

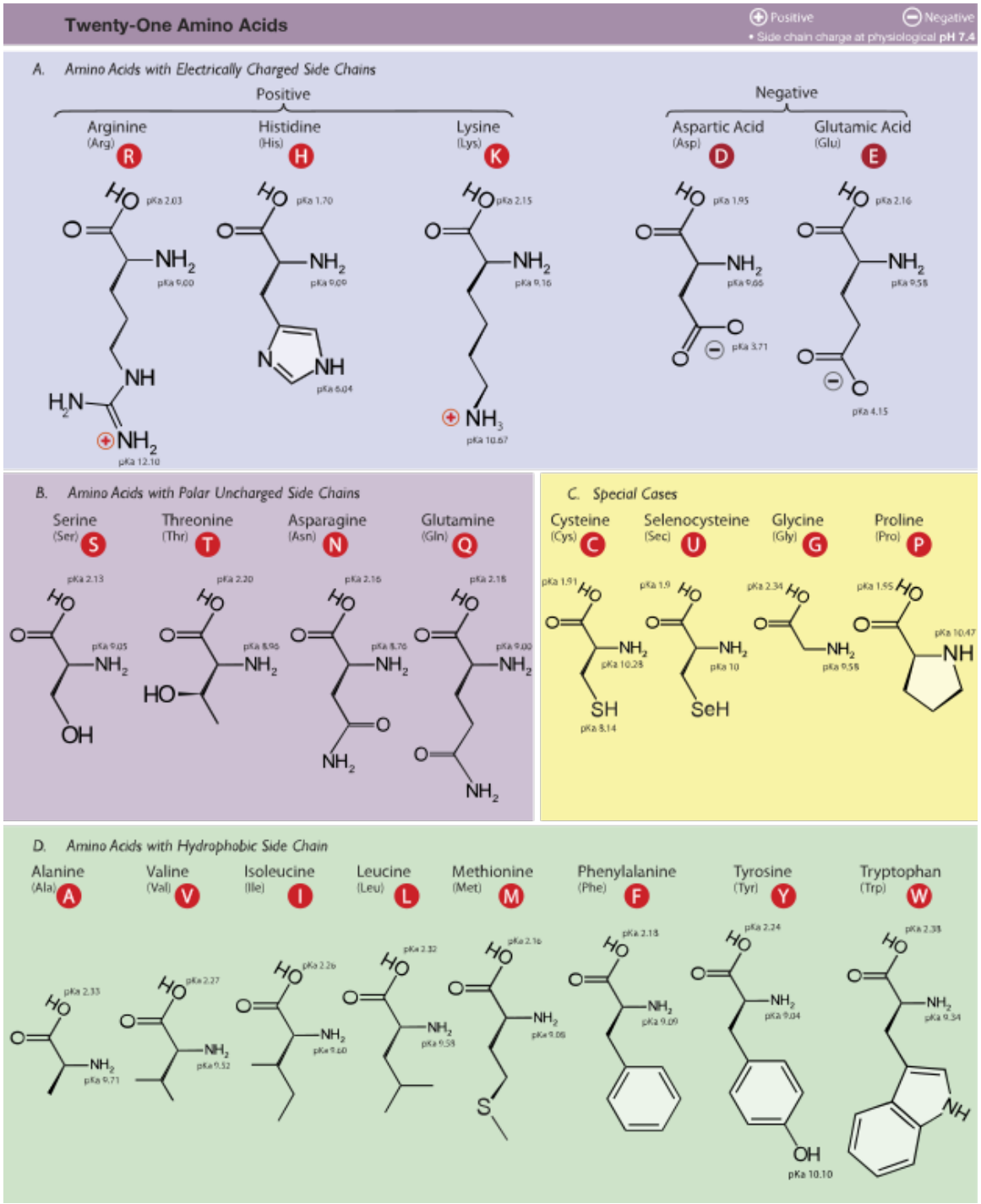
The genetic code

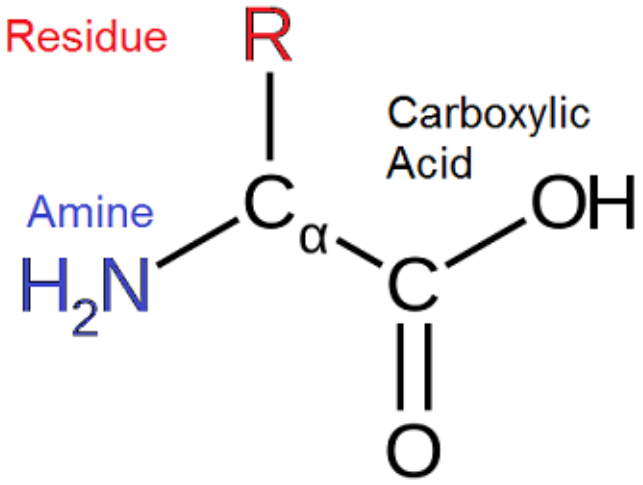
Standard genetic code

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

Wikipedia

Highly redundant: only 20 (or 21) amino acids formed from $4^3 = 64$ possible combinations.





By Dancojocari. https://commons.wikimedia.org/w/index.php?curid=9176441

Proteins

- **Linear polymer of amino acids**, linked together by peptide bonds. Average size ≈ 200 amino acids, can be over 1000.

- To a large extent, **cells are made of proteins.**

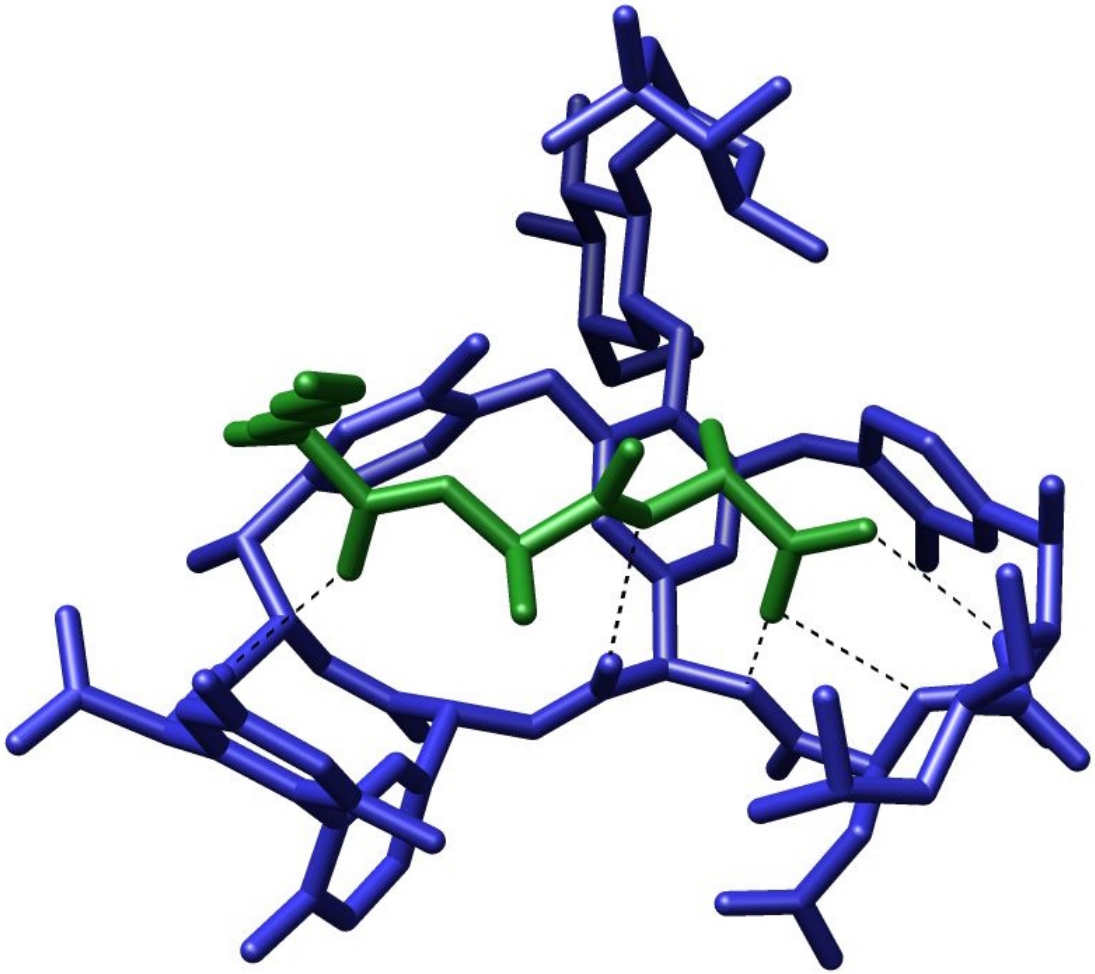
- Proteins determine **shape and structure of a cell**.
  Main instruments of **molecular recognition** and **catalysis**.

- **Complex structure** with four hierarchical levels.

  1. **Primary structure**: amino acid sequence.
  2. Different regions form locally regular **secondary structures** like α-*helices* and β-*sheets*.
  3. **Tertiary structure**: packing such structures into one or several 3D *domains*.
  4. Several domains arranged in a **quaternary structure**.

Molecular recognition

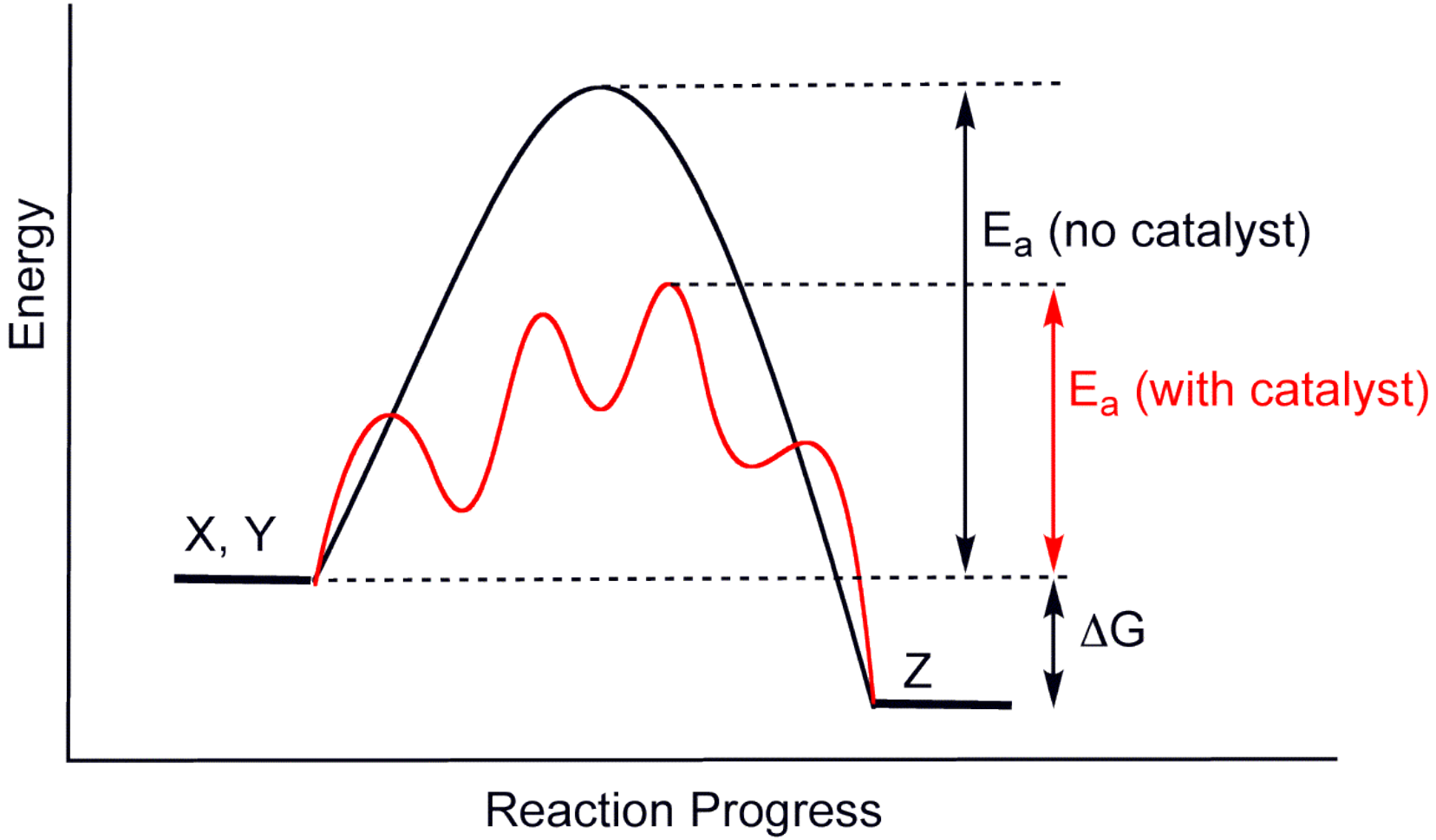
Interaction between molecules through noncovalent bonding



Crystal structure of a short peptide L-Lys-D-Ala-D-Ala (bacterial cell wall precursor) bound to the antibiotic vancomycin through hydrogen bonds. By M stone, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=2327682

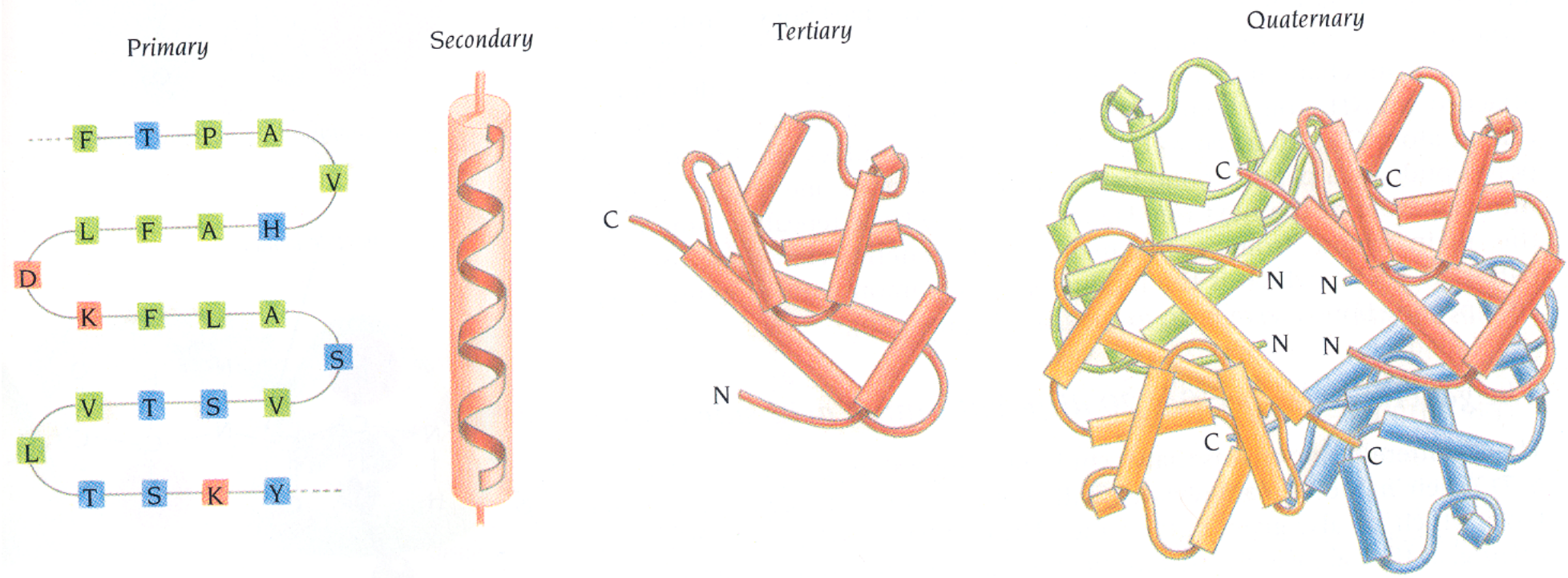
Catalysis

Increasing the rate of a chemical reaction by adding a substance $\rightsquigarrow$ catalyst.



Wikipedia

Protein Structure: primary to quaternary



Durbin et al., Cambridge University Press. https://doi.org/10.1017/CBO9780511790492.004

Structure is determined by the **primary sequence** and their **physico-chemical interactions** in the medium.

Structure determines functionality.

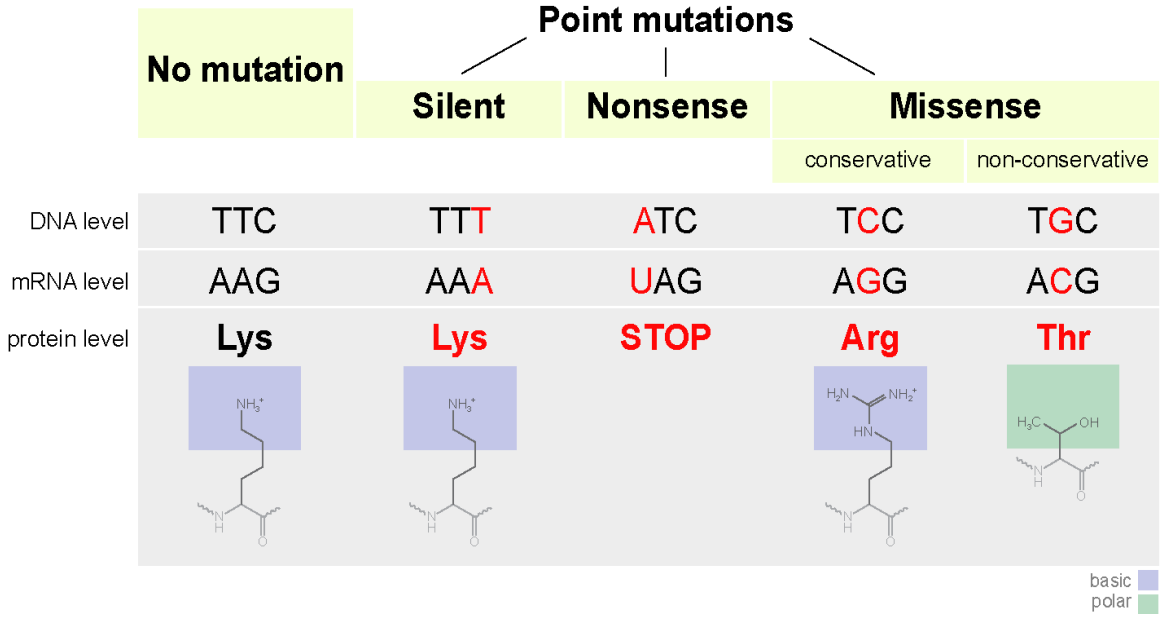
Mutations

- **Mutation**: Heritable change in the DNA sequence. Occur due to exposure to **ultra violet radiation** or other **environmental conditions**.

- **Two levels** at which a mutation can take place:
  - **Point mutation:** within a single gene.
    - **substitution** (change of one nucleotide),
    - **insertion** (addition of nucleotides),
    - **deletion**.
  - **Chromosomal mutation:** whole segments interchange, either on the same chromosome, or on different ones.

Point Mutations

- May arise from **spontaneous mutations** during **DNA replication.**
- The rate of mutation increased by **mutagens** (physical or chemical agent that changes the genetic material).
- Mutagens: Physical (UV-, X-rays or heat), or chemical (molecules misplace base pairs / disrupt helical shape of DNA).



Wikipedia. By Jonsta247 - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=12481467

Importance of Mutations

- Mutations are responsible for **inherited disorders & diseases.**
  Sickle-cell anemia caused by missense point mutation in **hemoglobin** (in blood cells, responsible for oxygen transport.)
  Hydrophilic **glutamic acid** replaced with hydrophobic **valine.**
  $\rightsquigarrow$ deformed red blood cells.

  Sequence for Normal Hemoglobin: 6th codon: **adenine** (A)

AUG	GUG	CAC	CUG	ACU	CCU	GAG	GAG	AAG	UCU	GCC	GUU	ACU
START	Val	His	Leu	Thr	Pro	Glu	Glu	Lys	Ser	Ala	Val	Thr

  Sickle Cell Hemoglobin: $\rightsquigarrow$ **thymine** (DNA), **uracil** (RNA)

AUG	GUG	CAC	CUG	ACU	CCU	GUG	GAG	AAG	UCU	GCC	GUU	ACU
START	Val	His	Leu	Thr	Pro	Val	Glu	Lys	Ser	Ala	Val	Thr

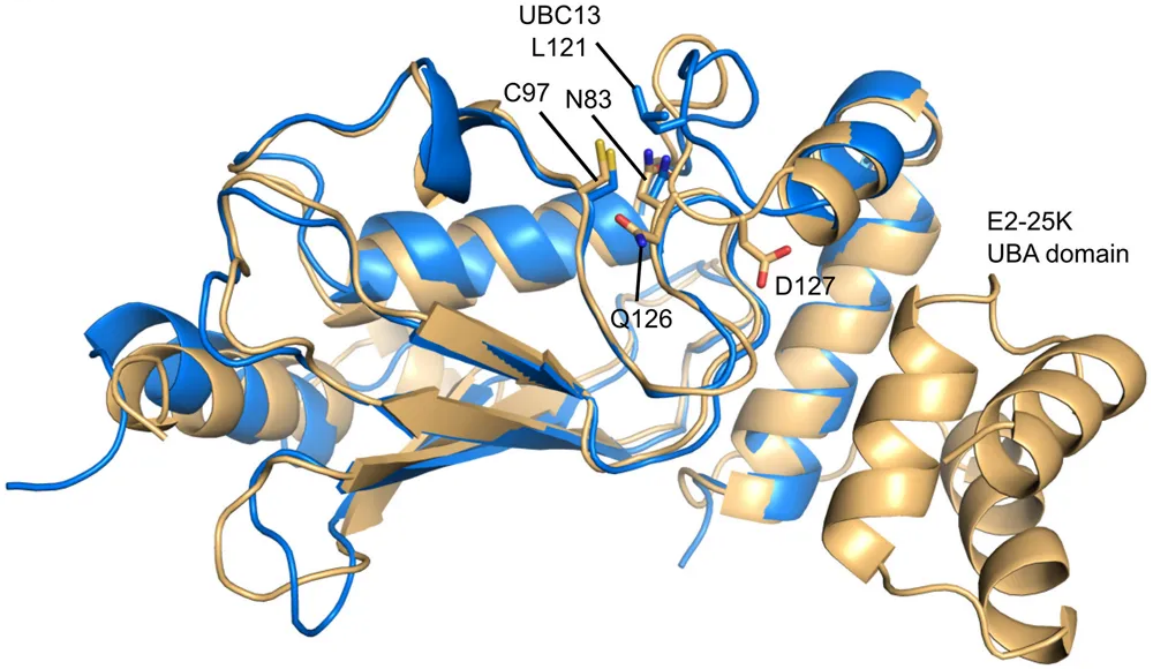
- Mutations are the source of **phenotypic variation**
  $\Rightarrow$ **new species** and **adaption** to environmental conditions.

Sequence Comparison: Motivation

Basic idea: **similar sequences** $\rightsquigarrow$ **similar proteins.**

Protein folding: 30 % sequence identity $\Rightarrow$ structures similar.

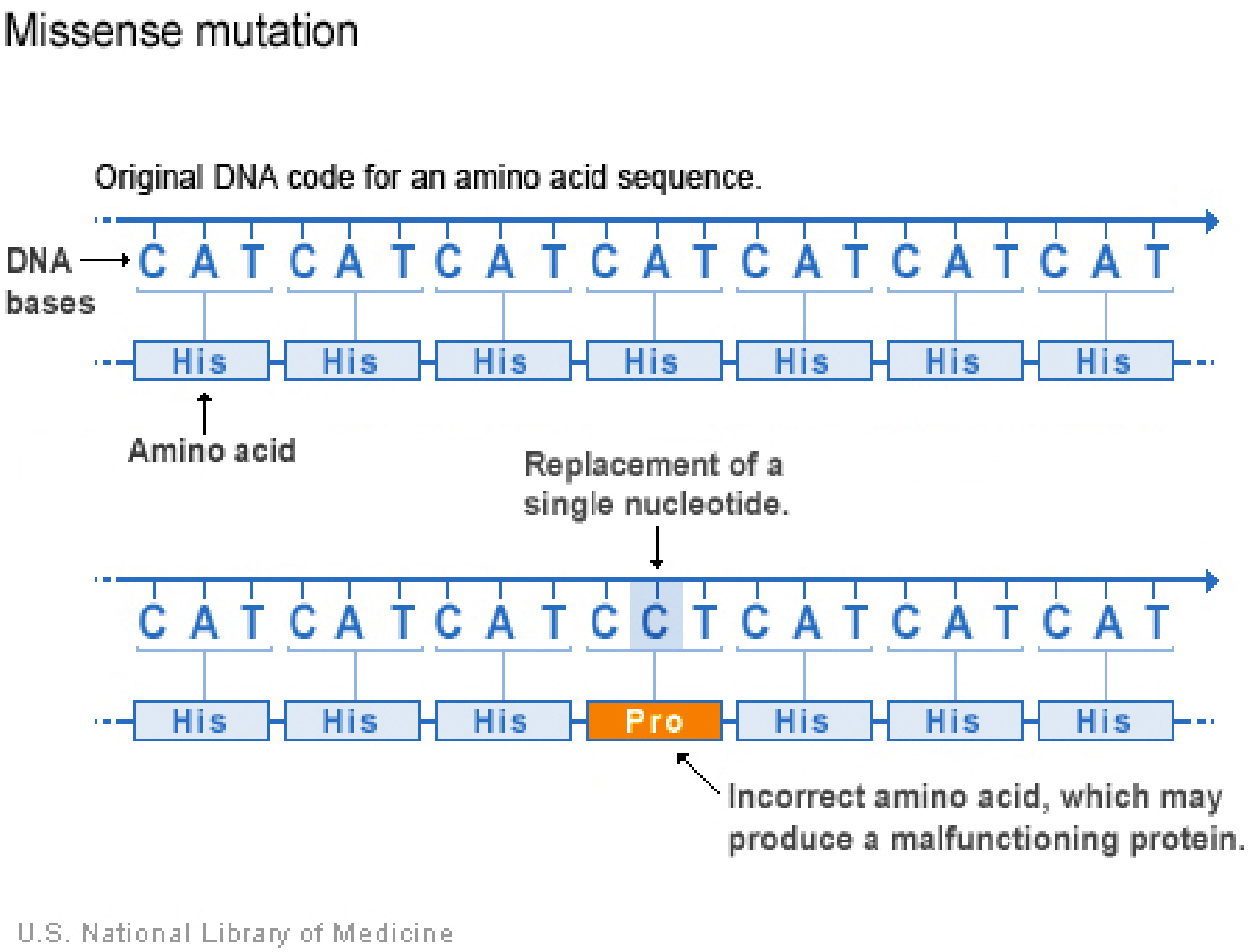
```
                    L1        L2        L3
E225K   1 MANIAVQRIKREFKEVLKSEETSKNQIKVDLVDENFTELRGEIAGPPDTPYEGGRYQLEI 60
UBC13   1 MAGLPRRIIK-ETQRLL---AEPVPGIKAEPDESNARYFHVVIAGPQDSPFEGGTFKLEL 56
            L4        L5 TT     L6     hhh     L7
E225K  61 KIPETYPFNPPKVRFITKIWHPNISSVTGAICLDILKDQWAAAMTLRTVLLSLQALLAAA 120
UBC13  57 FLPEEYPMAAPKVRFMTKIYHPNVDK-LGRICLDILKDKWSPALQIRTVLLSIQALLSAP 115
           L8
E225K 121 EPDDPQDAVVANQYKQNPEMFKQTARLWAHVYAGAPVSSPEYTKKIENLCAMGFDRNAVI 180
UBC13 116 NPDDPLANDVAEQWKTNEAQAIETARAWTRLYAMNNI---------------------- 152

E225K 181 VALSSKSWDVETATELLLSN 200
UBC13     --------------------
```



Rout et al., Scientific Reports, vol 8, no 7002 (2018)

Comparing sequences

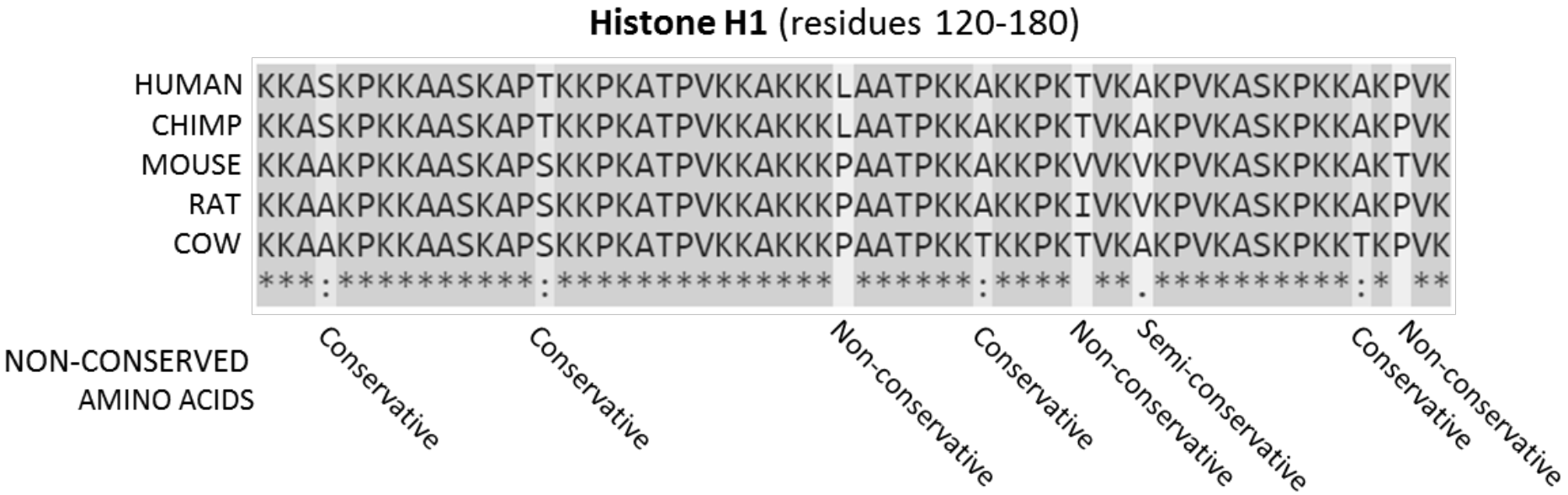
Theory: during evolution **mutations** occurred, creating differences between families of contemporary species.



https://commons.wikimedia.org/w/index.php?curid=25399199

Comparing sequences

Comparing two sequences: looking for **evidence** that they have **diverged from a common ancestor** by a **mutation process**.



By Thomas Shafee - Own work, CC BY 4.0, https://commons.wikimedia.org/w/index.php?curid=37188728

Sequence Alignment

Informal definition:
Alignment of sequences $x = x_1 \dots x_n$ and $y = y_1 \dots y_m$:
(i) **insert spaces**,
(ii) place resulting sequences **one above the other** so that every character or space has a counterpart.

Example: `ACBCDDDB` and `CADBDAD`. Possible alignments:

```
A C - - B C D D D B
  |     |   |   |
- C A D B - D A D -

- A C B C D D D B
  |   |     |
C A D B D A D - -
```

Optimal Alignment

Given: two sequences x and y over alphabet $\mathcal{A}$.

$\mathcal{A} = \{\mathtt{A}, \mathtt{G}, \mathtt{C}, \mathtt{T}\}$ (DNA)
$\mathcal{A} = \{\mathtt{A}, \mathtt{R}, \mathtt{N}, \mathtt{D}, \mathtt{C}, \mathtt{Q}, \mathtt{E}, \mathtt{G}, \mathtt{H}, \mathtt{I}, \mathtt{L}, \mathtt{K}, \mathtt{M}, \mathtt{F}, \mathtt{P}, \mathtt{S}, \mathtt{T}, \mathtt{W}, \mathtt{Y}, \mathtt{V}\}$ (proteins)

Formalizing **optimality of an alignment**: define

- the costs for **substituting** a letter by another letter
  $\Rightarrow$ **substitution matrix**;

- the costs for **insertion** $\Rightarrow$ **gap penalties**.

The Scoring Model

- **Idea:** assign a score to each alignment, choose best one.

- **Additive** scoring scheme: Total score = sum of all scores for pairs of letters + costs for gaps.
  Implicit assumption:
  Mutations at different sites have occurred **independently**.
  (In most cases) reasonable for DNA and protein sequences.

- **All** common algorithms use **additive scoring schemes.**

- Modeling dependencies is possible, but at the price of significant computational complexities.

Substitution Matrices

- **Expectation:**
  Identities in real alignments are more likely than by chance.

- Derive score for aligned pairs from a **probabilistic model.**

- **Score:** relative likelihood that two sequences are evolutionary related as opposed to being unrelated
  $\rightsquigarrow$ **score = ratio of probabilities.**

- **First assumption:** Ungapped alignment, $n = m$.

- R: **Random model:**
  Letter a occurs **independently** with some frequency q_a
  $\Rightarrow$ Pr(two sequences) = product of probabilities for each letter:

$$P(x, y|R) = \prod_i q_{x_i} \prod_i q_{y_i}.$$

Substitution Matrices

- M (**match**): aligned pairs occur with **joint probability**

$$P(x, y|M) = \prod_i p_{x_i y_i}$$

- Ratio $\rightsquigarrow$ **"odds ratio"**:

$$\frac{P(x, y|M)}{P(x, y|R)} = \prod_i \frac{p_{x_i y_i}}{q_{x_i} q_{y_i}}$$

- To arrive at an **additive** scoring system $\rightarrow$ **log-odds ratio**:

$$S = \sum_i \log\left(\frac{p_{x_i y_i}}{q_{x_i} q_{y_i}}\right) = \sum_i s(x_i, y_i)$$

- $s(a, b)$: log-likelihood ratio of pair (a, b) occurring as an **aligned pair** as opposed to an **unaligned pair** $\rightsquigarrow$ **substitution matrix**.

BLOSUM62 substitution matrix

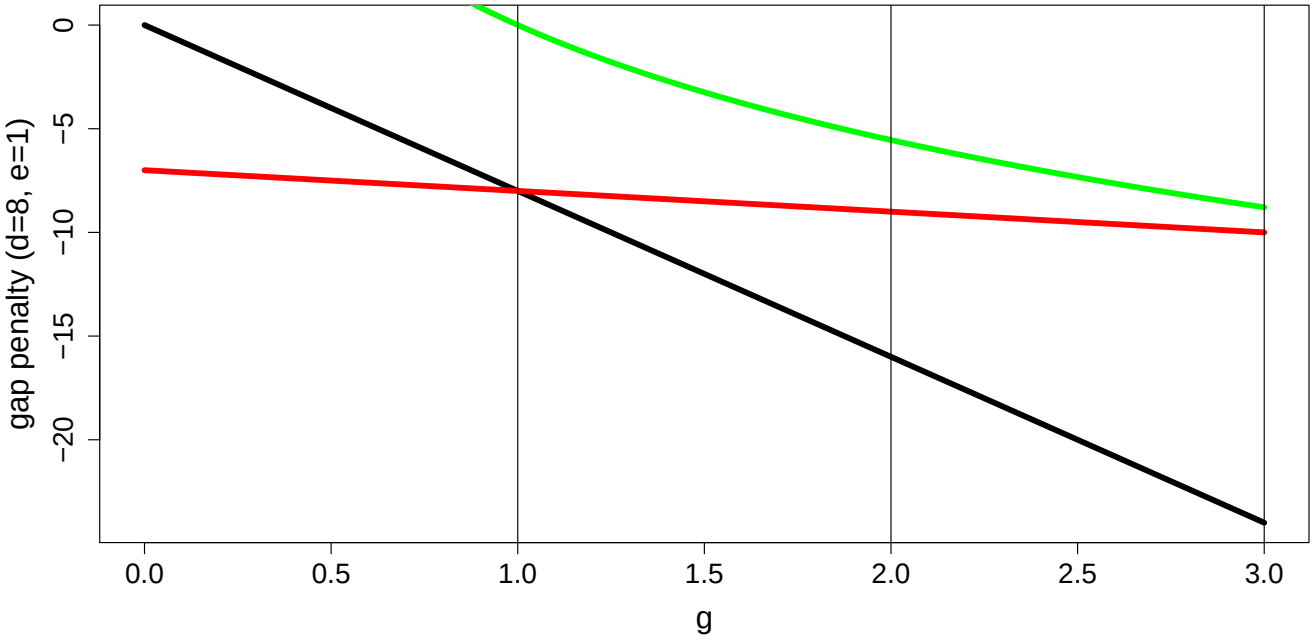
	Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val
Ala	4																			
Arg	−1	5																		
Asn	−2	0	6																	
Asp	−2	−2	1	6																
Cys	0	−3	−3	−3	9															
Gln	−1	1	0	0	−3	5														
Glu	−1	0	0	2	−4	2	5													
Gly	0	−2	0	−1	−3	−2	−2	6												
His	−2	0	1	−1	−3	0	0	−2	8											
Ile	−1	−3	−3	−3	−1	−3	−3	−4	−3	4										
Leu	−1	−2	−3	−4	−1	−2	−3	−4	−3	2	4									
Lys	−1	2	0	−1	−3	1	1	−2	−1	−3	−2	5								
Met	−1	−1	−2	−3	−1	0	−2	−3	−2	1	2	−1	5							
Phe	−2	−3	−3	−3	−2	−3	−3	−3	−1	0	0	−3	0	6						
Pro	−1	−2	−2	−1	−3	−1	−1	−2	−2	−3	−3	−1	−2	−4	7					
Ser	1	−1	1	0	−1	0	0	0	−1	−2	−2	0	−1	−2	−1	4				
Thr	0	−1	0	−1	−1	−1	−1	−2	−2	−1	−1	−1	−1	−2	−1	1	5			
Trp	−3	−3	−4	−4	−2	−2	−3	−2	−2	−3	−2	−3	−1	1	−4	−3	−2	11		
Tyr	−2	−2	−2	−3	−2	−1	−2	−3	2	−1	−1	−2	−1	3	−3	−2	−2	2	7	
Val	0	−3	−3	−3	−1	−2	−2	−3	−3	3	1	−2	1	−1	−2	−2	0	−3	−1	4

Wikipedia

Gap penalties

Gap penalty types for a gap of length g:

- **Linear**: $\gamma(g) = -gd$, with d being the **gap weight**.
- **Affine**: $\gamma(g) = -d - (g-1)e$,
  gap-open penalty d, **gap-extension** penalty e. Usually $e < d$.
- **Convex**: e.g. $\gamma(g) = -d\log(g)$. Each additional space contributes less than the previous space.



Global Alignment: Needleman-Wunsch algorithm

The Global Alignment problem:
INPUT: two sequences $x = x_1 \dots x_n$ and $y = y_1 \dots y_m$.
TASK: Find optimal alignment for linear gap penalties $\gamma(g) = -gd$.

Let $F(i, j)$ be the optimal alignment score of the **prefix sequences** $x_{1\dots i}$ and $y_{1\dots j}$. A zero index $i = 0$ or $j = 0$ refers to an **empty sequence.** $F(i, j)$ has following properties:

Base conditions:

$$F(i, 0) = \sum_{k=1}^{i} -d = -id$$

$$F(0, j) = \sum_{k=1}^{j} -d = -jd, \quad F(0, 0) = 0.$$

Recurrence relation:

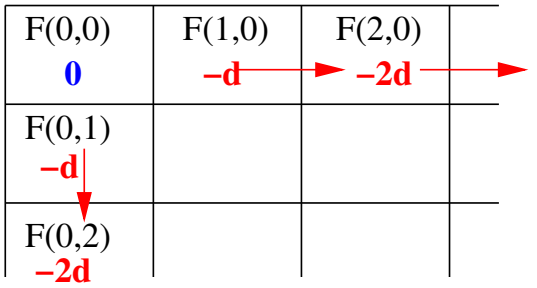
for $1 \le i \le n,\ 1 \le j \le m$:

$$F(i, j) = \max \begin{cases} F(i-1, j-1) + s(x_i, y_j) \\ F(i-1, j) - d \\ F(i, j-1) - d \end{cases}$$

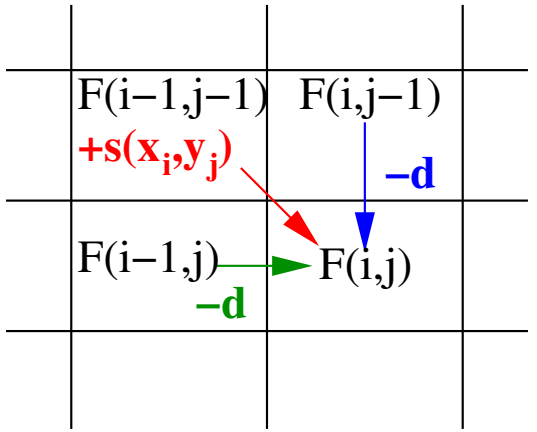
Tabular Computation of Optimal Alignment

Starting from $F(0,0) = 0$, fill the whole matrix $(F)_{ij}$:

for $i = 0$ or $j = 0$, calculate new value from left-hand (upper) value.



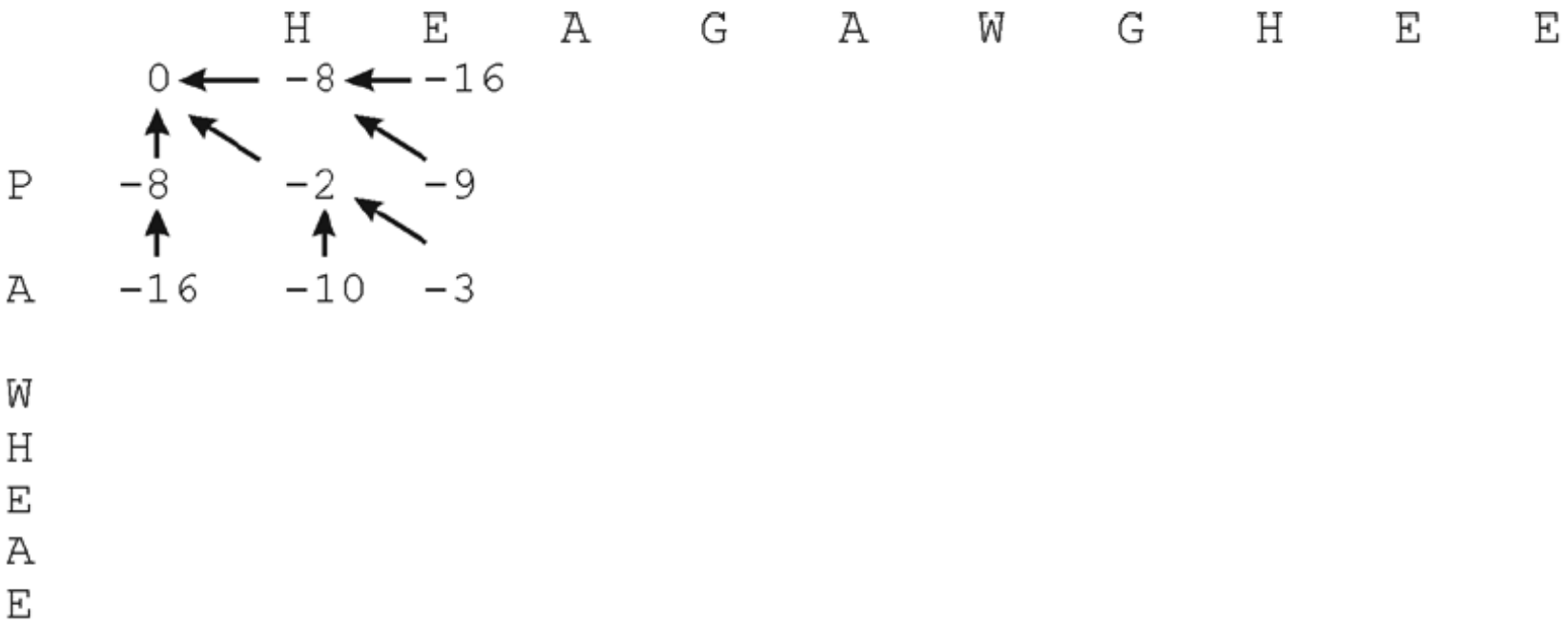
for $i, j \geq 1$, calculate the bottom right-hand corner of each square of 4 cells from one of the 3 other cells:



keep a pointer in each cell back to the cell from which it was derived $\Rightarrow$ **traceback pointer**.

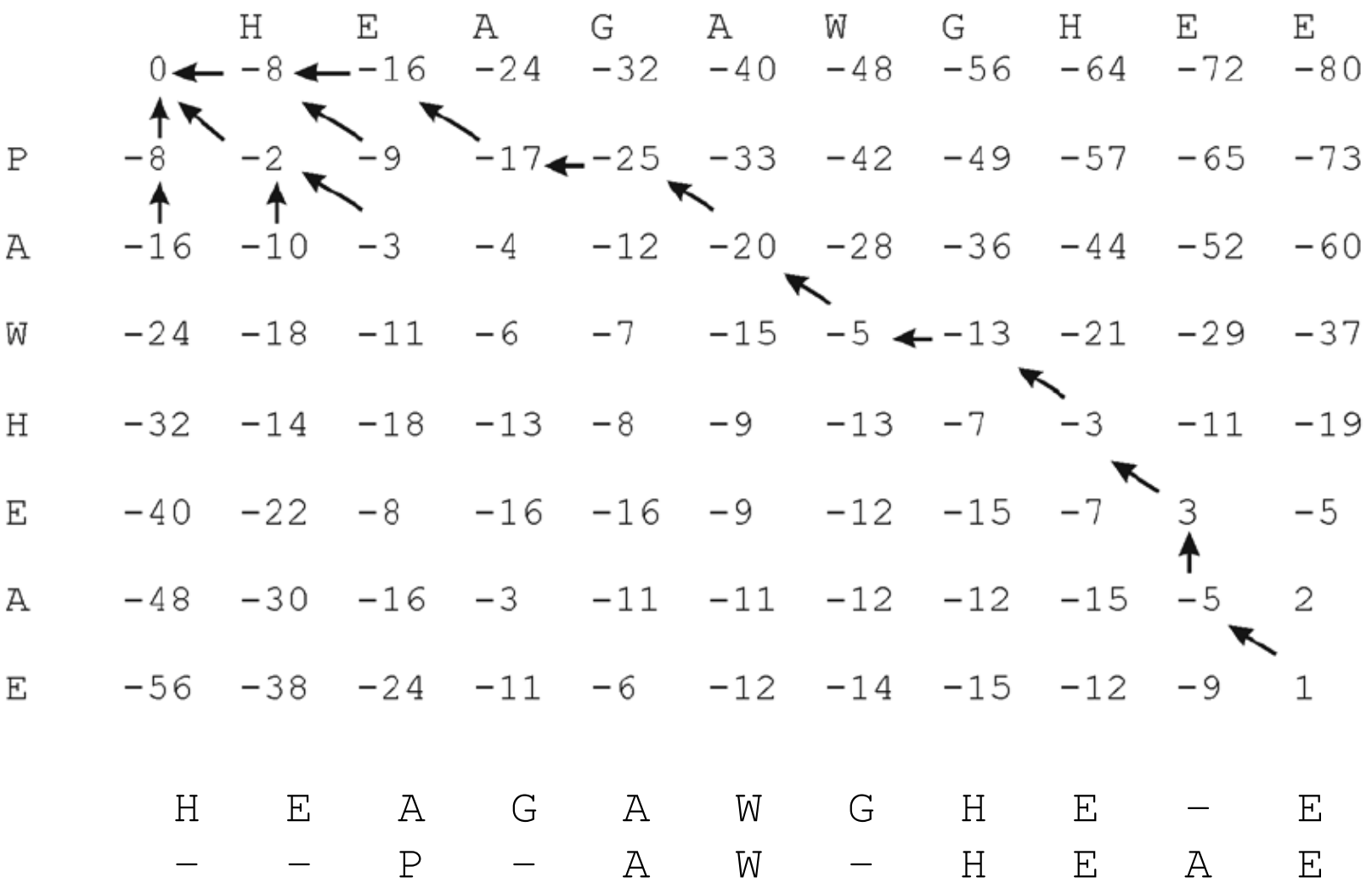
Global Alignment: Example

x = HEAGAWGHEE, y = PAWHEAE. Linear gap costs $d = 8$.
Scoring matrix: BLOSUM50



Durbin et al., Cambridge University Press. https://doi.org/10.1017/CBO9780511790492.004

Example: traceback procedure



Add pair of symbols: $\nwarrow$: (x_i, y_j), $\uparrow$: $(-, y_j)$, $\leftarrow$: $(x_i, -)$

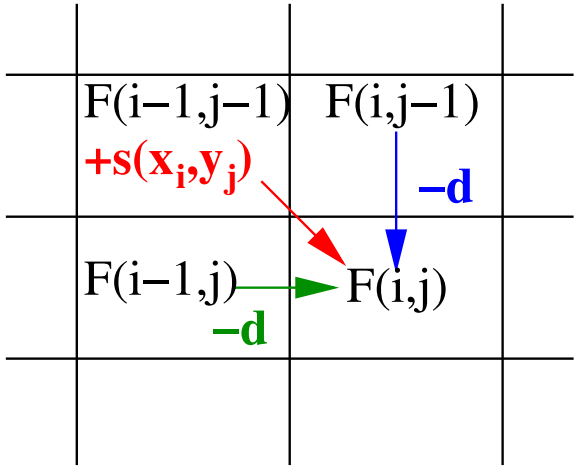
Adapted from Durbin et al., Cambridge University Press. https://doi.org/10.1017/CBO9780511790492.004

Time and Space Complexity

Theorem 1. *The time complexity of the Needleman-Wunsch algorithm is $O(nm)$. Space complexity is $O(m)$, if only $F(x, y)$ is required, and $O(nm)$ for the reconstruction of the alignment.*

Proof:

Time: when computing $F(i, j)$, only cells $(i-1, j-1)$, $(i, j-1)$, $(i-1, j)$ are examined $\rightsquigarrow$ constant time. There are $(n+1)(m+1)$ cells $\rightsquigarrow$ $O(nm)$ **time complexity.**

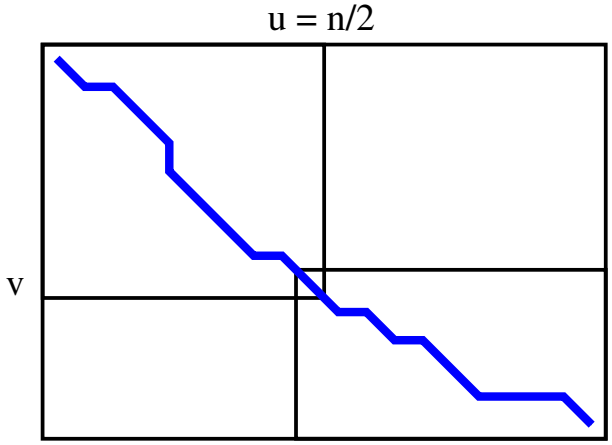


Space : row-wise computation of the matrix: for computing row k, only row $k-1$ must be stored $\rightsquigarrow$ $O(m)$ **space.**

Reconstructing the alignment: all traceback pointers must be stored $\rightsquigarrow$ $O(nm)$ **space complexity.**

Global Alignment in Linear Space

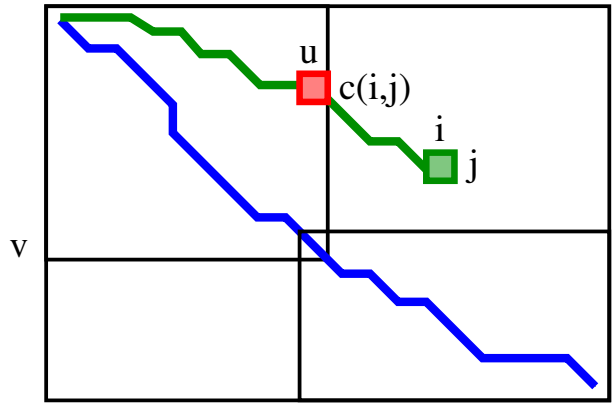
- Problem: genomic scale sequence analysis: comparing two large genomic sequences: $m, n \approx 10^6 \Rightarrow$ space complexity 10^{12} is clearly unacceptable!
- **Solution:** linear space algorithms with space complexity $O(m+n)$.
- Basic idea: **divide and conquer**. Let $u = \lfloor \frac{n}{2} \rfloor$ be the integer part of $\frac{n}{2}$.
  - Let v be a row index such that the cell (u, v) is on the optimal alignment.
  - Split dynamic programming problem into two parts:
    $(0,0) \rightarrow (u,v)$ and $(u,v) \rightarrow (n,m)$.
    Optimal alignment will be concatenation of individual sub-alignments.
  - Repeat splitting until until $u = 0$: trivial



Question: how can we find v?

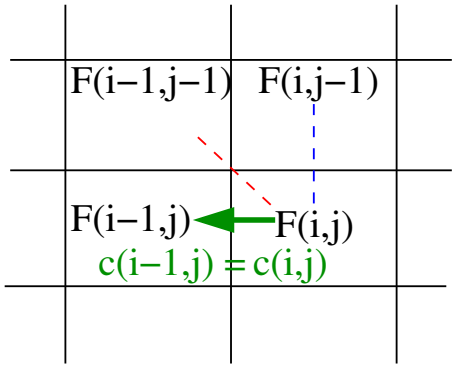
Global Alignment in Linear Space

- For $i \geq u$ define $c(i,j)$ such that $(u, c(i,j))$ is on the optimal path from $(1,1) \rightarrow (i,j)$.



- Let (i', j') be the preceding cell to (i,j) from which $F(i,j)$ is derived. Update $c(i,j)$ as:

$$c(i,j) = \begin{cases} j & \text{, if } i = u, \\ c(i',j') & \text{, else} \end{cases}$$



- Local operation $\rightsquigarrow$ need to store only the previous row of $c()$.
- Finally, $v = c(n,m)$.

Global Alignment in Linear Space: Example

Computing the c matrix for the first step (i = n = 6, j = m = 4, u = 3).

The c values are written as subscripts. BLOSUM62, linear gap costs $d = 8$.

		0		1		2		3		4		5		6
		•		H		E		A		G		A		W
0	•	0	←	-8	←	-16	←	-24_0	←	-32_0	←	-40_0	←	-48_0
		↑	↖		↖		↖				↖			
1	P	-8		-2		-9		-17_1	←	-25_1		-33_0	←	-41_0
		↑	↖	↑	↖		↖				↖			
2	A	-16		-10		-3		-4_2	←	-12_2		-20_1	←	-28_1
		↑		↑			↖		↖		↖		↖	
3	W	-24		-18		-11		-6_3		-7_2		-15_2		-5_1
		↑	↖		↖		↖		↖		↖			↑
4	H	-32		-14		-18		-13_4		-8_3		-9_2		-13_1

Every $c(i, j)$ defines a row index v such that $(u, c(i, j))$ is on the optimal path from $(1, 1)$ to $(i, j) \rightsquigarrow v = c(6, 4) = 1$, so (3,1) is our desired element on the optimal path form (1,1) to (6,4).

Local Alignments

The Local Alignment problem:
INPUT: two sequences $x = x_1, \ldots, x_n$ and $y = y_1, \ldots, y_m$.
TASK: find subsequences a of x and b of y,
whose similarity (=optimal global alignment score) is maximal
(over all such pairs of subsequences).
Assume linear gap penalties $\gamma(g) = -gd$.

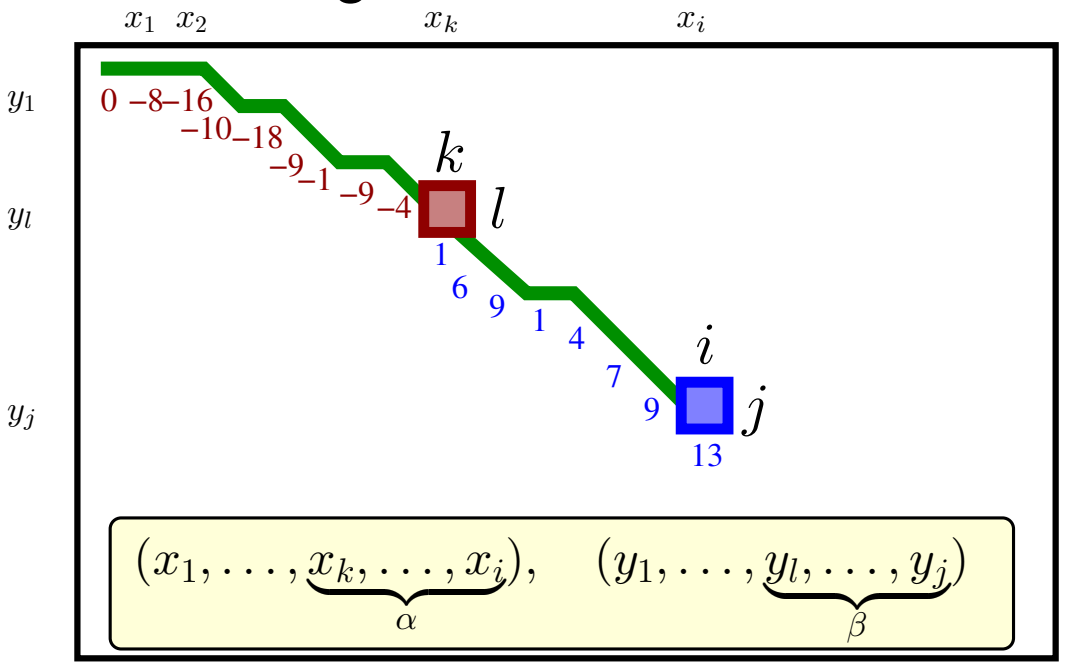
Subsequence = **contiguous** segment of a sequence.

Consider first a simpler problem by **fixing the endpoint** of the subsequences at index pair (i, j):
Local suffix alignment problem: given x, y, i, j, find suffixes α of $x_{1,\ldots,i}$ and β of $y = y_{1,\ldots,j}$ such that their global alignment score is maximal.

$$(x_1, \ldots, \underbrace{x_k, \ldots, x_i}_{\alpha}), \quad (y_1, \ldots, \underbrace{y_l, \ldots, y_j}_{\beta})$$

Local suffix alignments

Consider global alignment path to cell (i, j). Where to start?
Intuition: Indices (k, l) found by following the path back to $(0, 0)$, but stopping at the first negative value.



Remark: If we consider all solutions (i.e. for all (i, j) pairs), we look at all possible subsequences (no restrictions on α, β)

Maximal solution of local suffix alignment over all pairs (i, j)
= solution of local alignment problem.

Smith-Waterman Algorithm

$F(i, j)$: optimal local suffix alignment for indices i, j.

Global alignment with one **modification:**
Prefixes whose scores are ≤ 0 are **discarded**
$\rightsquigarrow$ alignment can **start anywhere**.

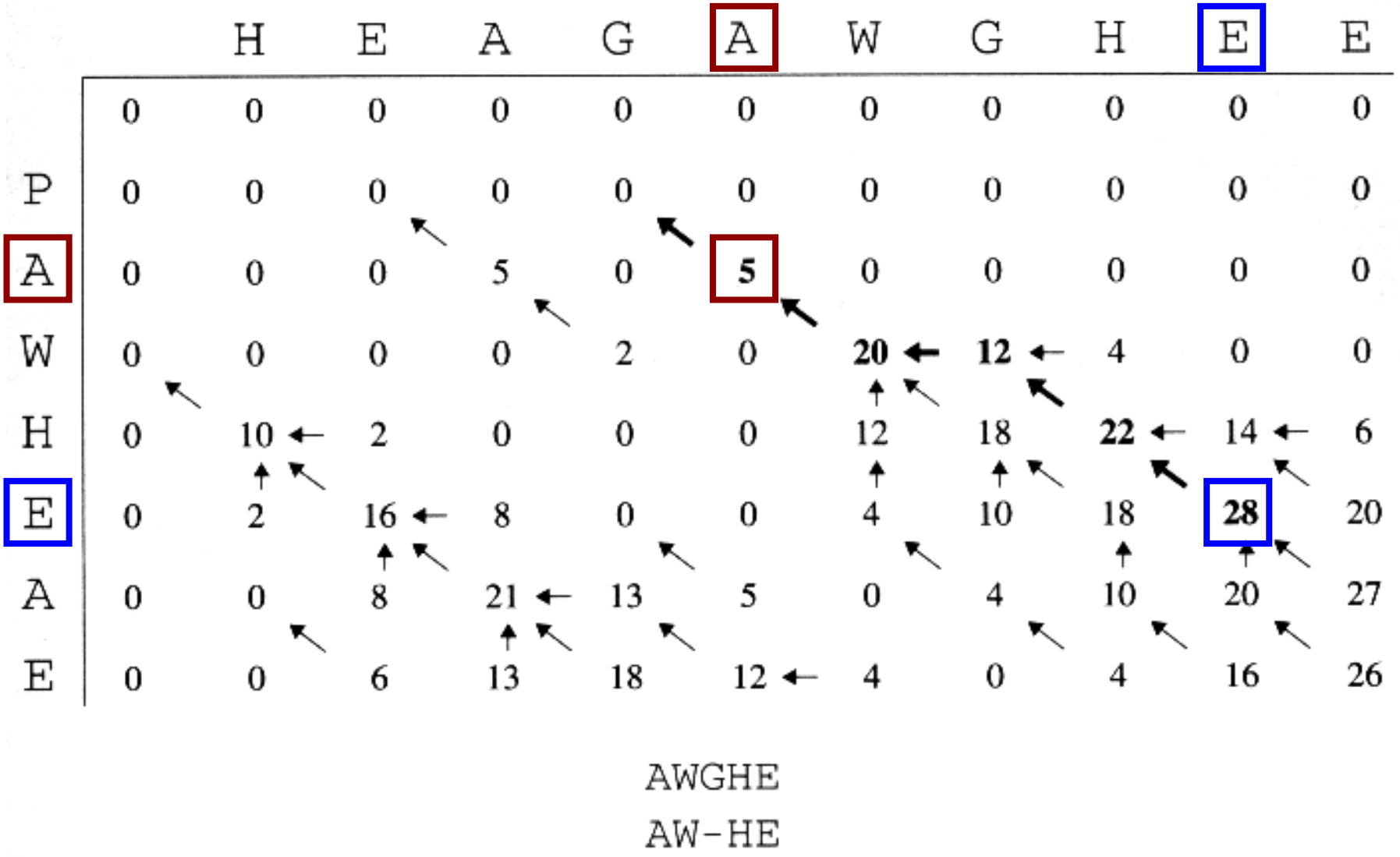
Recurrence relation:
$$F(i,j) = \max \begin{cases} 0 \\ F(i-1, j-1) + s(x_i, y_j) \\ F(i-1, j) - d \\ F(i, j-1) - d \end{cases}$$

Finally, find indices i^* and j^* **after which the similarity only decreases.** Stop the alignment there.

$$F(i^*, j^*) = \max_{i,j} F(i,j)$$

Traceback...

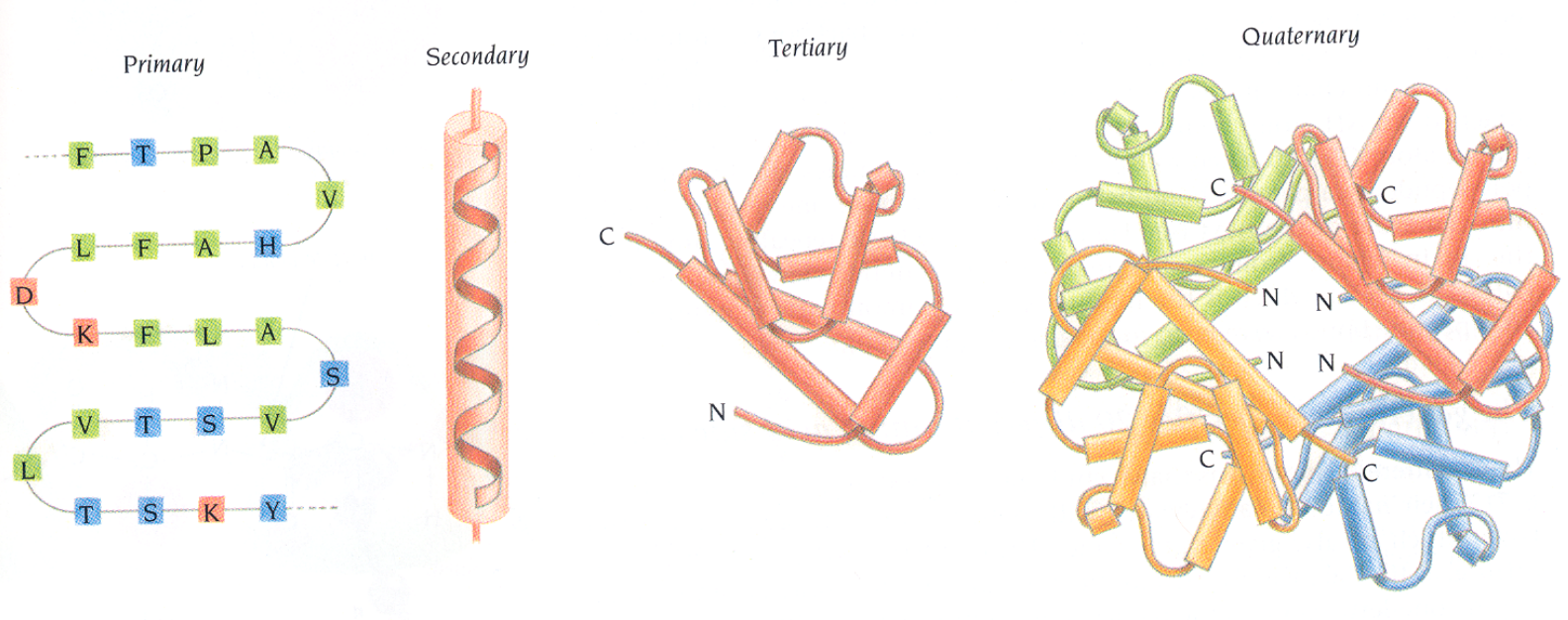
...starts at highest value until a cell with 0 is reached.



Adapted from Durbin et al., Cambridge University Press. https://doi.org/10.1017/CBO9780511790492.004

Local vs. Global Alignment: Biological Considerations

- Many proteins have **multiple domains**, or modules.
- Some domains are present (with high similarity) in many other proteins
- **Local** alignment can detect similar regions in otherwise dissimilar proteins.



Durbin et al., Cambridge University Press. https://doi.org/10.1017/CBO9780511790492.004

Other gap models

- **So far:** linear gap model. Not ideal for biological sequences, since it penalizes additional gap steps as much as the first. But in reality: When gaps do occur, they are often longer than one character.

```
HBA_HUMAN  GSAQVKGHGKKVADALTNAVAHV---D--DMPNALSALSDLHAHKL
           ++ ++++H+ KV    + +A  ++           +L+ L+++H+ K
LGB2_LUPLU NNPELQAHAGKVFKLVYEAAIQLQVTGVVVTDATLKNLGSVHVSKG
```

Durbin et al., Cambridge University Press. https://doi.org/10.1017/CBO9780511790492.004

- For a **general gap cost function** $\gamma(g)$, we can still use the standard dynamic programming recursion with slight modifications:

$$F(i,j) = \max \begin{cases} F(i-1,j-1) + s(x_i, y_j) \\ F(k,j) + \gamma(i-k), \quad k = 0, \ldots, i-1, \\ F(i,k) + \gamma(j-k), \quad k = 0, \ldots, j-1. \end{cases}$$

- **Problem:** requires $O(n^3)$ operations to align two sequences of length n, rather than $O(n^2)$. Why? $\rightsquigarrow$ exercises...

Alignment with affine gap costs

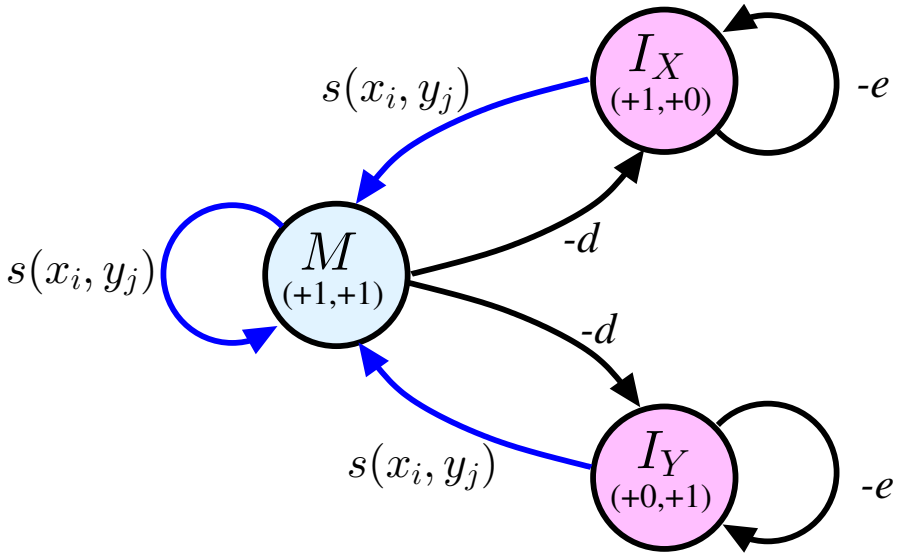
For affine gap costs, $\gamma(g) = -d - (g-1)e$, there exists a **solution:**
Modify recurrence by introducing another two "states". Denote by

- $M(i,j)$ the best score given that x_i is aligned to y_j,
- $I_x(i,j)$ the best score given that x_i is aligned to a gap,
- $I_y(i,j)$ the best score given that y_j is aligned to a gap.

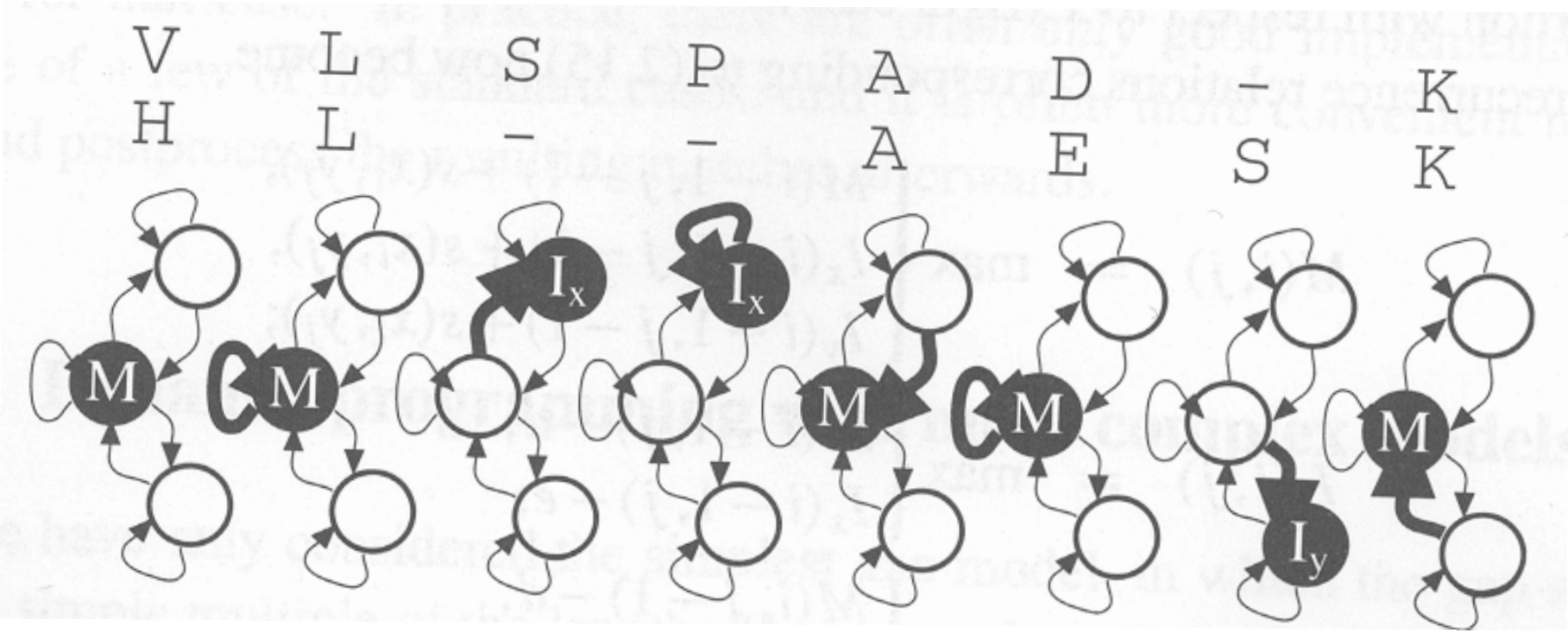
$$M(i,j) = \max \begin{cases} M(i-1,j-1) + s(x_i, y_j) \\ I_x(i-1,j-1) + s(x_i, y_j) \\ I_y(i-1,j-1) + s(x_i, y_j) \end{cases}$$

$$I_x(i,j) = \max \begin{cases} M(i-1,j) - d \\ I_x(i-1,j) - e \end{cases}$$

$$I_y(i,j) = \max \begin{cases} M(i,j-1) - d \\ I_y(i,j-1) - e \end{cases}$$



Example FSA alignment



Durbin et al., Cambridge University Press. https://doi.org/10.1017/CBO9780511790492.004

FSA alignment corresponds to path through states.

Probabilistic version $\rightsquigarrow$ **Hidden Markov models** (next chapter)