CMSC 423:
Bioinformatics Algorithms

Fall 2025

UNIVERSITY OF
MARYLAND

Course Info

Instructor: Rob Patro (nomad@umd.edu)

Office: 3220 IRB

Website: https://umd-cmsc423.github.io/f2025/

Course Info

TA:

Rachel Parsons
email: rparsons@umd.edu
Office hours: Thurs. 3 - 4 PM (AVW 4140)

Course Info

ADS: https://www.counseling.umd.edu/ads/

Academic Integrity: https://academiccatalog.umd.edu/undergraduate/registration-academic-requirements-regulations/academic-integrity-student-conduct-codes/

Piazza Page: https://piazza.com/umd/fall2025/cmsc423

Gradescope: https://www.gradescope.com/courses/1101775

If you have a class-related e-mail: Please **prefix the subject with [CMSC423_F25]**, so that my filter will pick it up and it won't be accidentally routed to SPAM.

Coursework & Grading

Coursework and grading: The coursework will consist of a number of different programming projects, a midterm exam and a final exam. The breakdown of weights for these different assignments will be as follows:

- Programming assignments - 50% (Currently plan for 6 but may only fit 5)
- Midterm 1 - 12%
- Midterm 2 - 13%
- Final Exam — 25%

Late Policy

- The late penalty for assignments is 1% per hour late up to a maximum of 48 hours.
- The points are deducted in integer amounts and rounded down (i.e. 59 minutes late is a 0% penalty).
- You are allowed 1 "free" late submission (up to 48 hours) over the course of all assignments.
- When you apply your "free" late pass, you will not lose any additional points for late turn in of that assignment.
- You must specify that you are using your "free" late pass when you submit your assignment (e-mail both the TA and me).
- Your use of the "free" late is a one-time, irrevocable decision (you can't switch which project you apply it to).

Regrade policy: All requests to re-grade, re-check, or re-mark an assignment or exam question **must be made in writing**. When the assignment is re-graded, it will be re-checked in its entirety. This means that *it is possible to lose points on other problems if they were graded incorrectly or too leniently the first time*. Therefore, I urge you to thoroughly consider each regrade request you make.

Academic Integrity maintain it!

TLDR : Don't cheat. Don't copy code from friends, classmates, or the internet for the short programming assignments or the projects. Don't provide code to classmates for any of the assignments or projects. Don't cheat on the exams. Be cool, and everything will be cool.

Academic integrity is a very serious issue. Any assignment, project or exam you complete in this course is expected to be your own work. If you are allowed to discuss the details of or work together on an assignment, this will be made explicit. Otherwise, you are expected to complete the work yourself. *Plagarism is not just the outright copying of content*. If you paraphrase someone else's thoughts, words, or ideas and you don't cite your source, this constitues plagraism. It is always much better to turn in an incorrect or incomplete assignment representing your own efforts than to attempt to pass off the work of another as your own. **If you are academically dishonest in this course, you will recieve a grade of XF, and you will be reported to the university's Office of Student Conduct**.

Textbooks

Based on student feedback from previous offerings of this course, there *is no required text.*

Additional material will be made available on the course website as needed.

However, this is an *upper-level* course, and you should *absolutely* seek out other sources explaining these topics from different angles, using different notations and examples, etc.

If you seek out other sources and still are having difficulty with an idea, *please* reach out to us (myself & Rachel). Also, consider reaching out to your peers *via* Piazza.

Other Textbooks

Genomics algorithms, data structures, and statisitcal models:

- Bioinformatics Algorithms: An Active Learning Approach
- Genome Scale Algorithm Design (Mäkinen, Belazzougui, Cunial, Tomescu 2015)
- Biological Sequence Analysis (Durbin, Eddy, Krogh, Mitchinson 1998)

Basics of algorithms and data structures:

This course will assume familiarity with basic algorithms and data structures, though I will attempt to refresh everyone's memory on relevant concepts when we cover them. If you need a refresher on algorithmic basics, I recommend the following resources:

- Algorithms (Dasgupta, Papadimitriou, and Vazirani 2006)
- Algorithm Design (Kleinberg and Tardos 2006)
- Introduction to Algorithms, 3rd edition(Cormen, Leiserson, Rivest and Stein, 2009)

Molecular biology:

We will cover the basic required molecular Biology in the course. However if you're not familiar with basic molecular Biology, there are some useful resources worth reading:

- Molecular Biology of the Cell (Alberts, Johnson, Lewis, Raff, Roberts and Walter, 2002)
- Molecular Biology: Principles of Genome Function 2nd Edition (Craig, Green, Greider, Storz, Wolberger, Cohen-Fix, 2014)
- Molecular Biology (Clark and Pazdernik 2012)

Software development and the command line

While you should have had, in the prerequisites for this course, exposure to the relevant tools (e.g. how to properly create a tarball, how to invoke the compiler from the command line / a script), I realize it may have been a while since you have used these skills. If you feel you need a refresher on these topics, I would strongly suggest checking out "The missing semester" (an MIT course dedicated to these various miscellaneous topics).

Other Textbooks

Software development and the command line

While you should have had, in the prerequisites for this course, exposure to the relevant tools (e.g. how to properly create a tarball, how to invoke the compiler from the command line / a script), I realize it may have been a while since you have used these skills. If you feel you need a refresher on these topics, I would strongly suggest checking out "The missing semester" (an MIT course dedicated to these various miscellaneous topics).

The "workhorse" bioinformatics algorithms are implemented in tools that are *command line applications*. They follow (often) traditional Unix design principles, are invoked from the command line, read files as input and produce other files as output.

Hence, our projects will be designed to work in a similar way.

Basics you should refresh

- File and directory structure and navigation
- File movement, copying, compression (tar + gzip), etc.
- Working in a **compiled** language (C, C++, Go, Rust, Java, etc.)
- Building software **from the command line** (bash script, Makefile, CMake, Cargo, etc.)
- (If choosing Java): Making a Java program act as a "normal" executable
- Reading command line parameters (passing parameters and options to your program)
- Reading and writing of files (reading files from disk, writing output to disk)
- Serialization and Deserialization of data structures (saving data structures to disk & reading them back)

More syllabus stuff

Course Objectives

The main objective of this course will be to provide an understanding of some of the algorithms, data structures, and methods that underlie *modern* computational genomics. This course is intended as a broad introduction to bioinformatics and computational biology. However, this is a huge field, so we will not cover everything, and what we do cover will not all be at the same depth (e.g. we will spend more time discussing indexing than clustering). Our perspective will be a computational and algorithmic one, though we will take the time to understand the necessary biology and motivation for the problems we discuss. At the end of this course, you should have a good understanding of how new challenges in genomics drive algorithmic innovations and how algorithmic innovations enable new and improved biological analyses.

Some final notes about this course

A few notes about what you should expect from CMSC423 this semester:

I've noticed that this course has a *little bit* of a reputation to be challenging (esp. when I teach it).

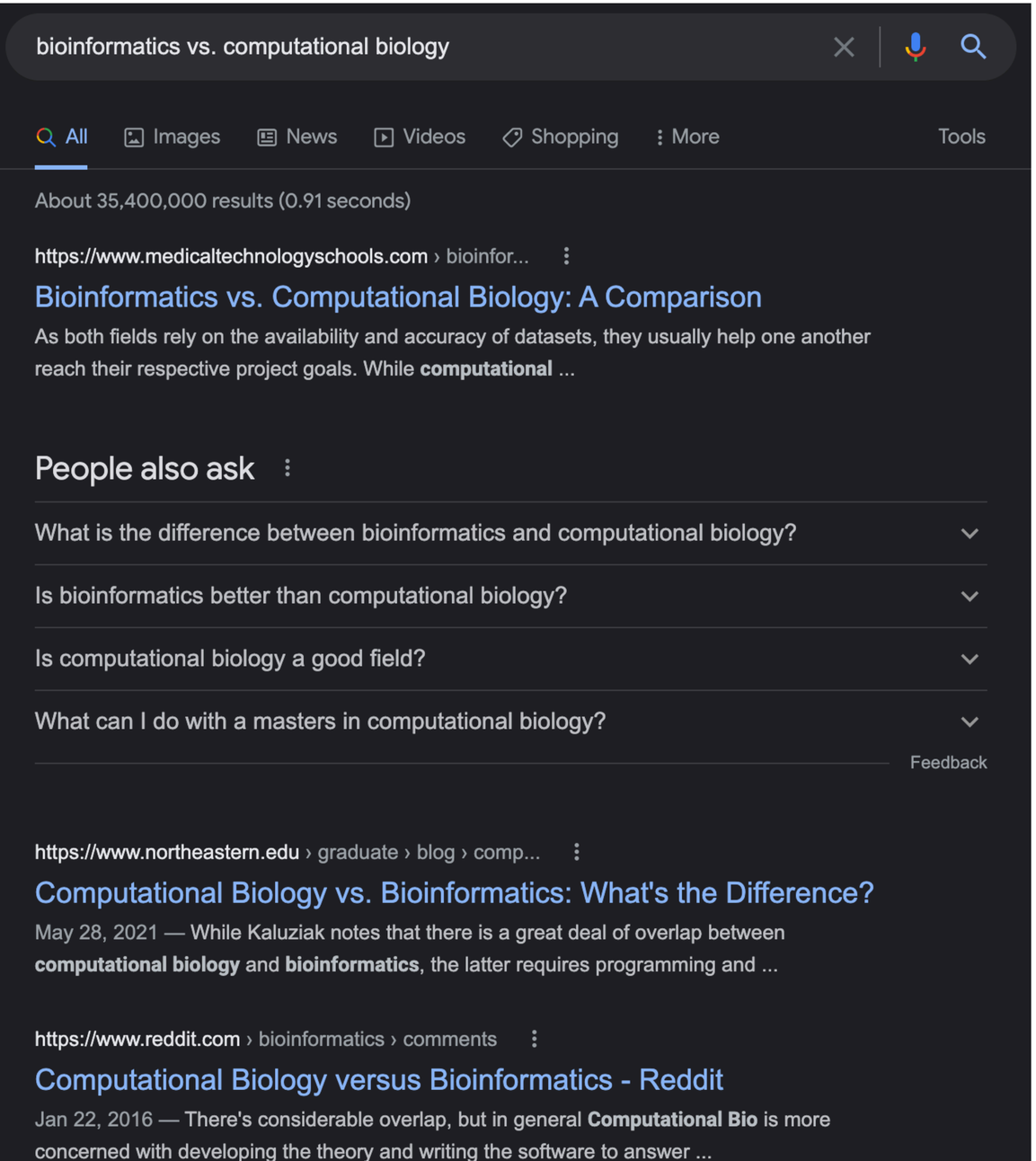
However, this course isn't designed to be a crazy hard course etc.

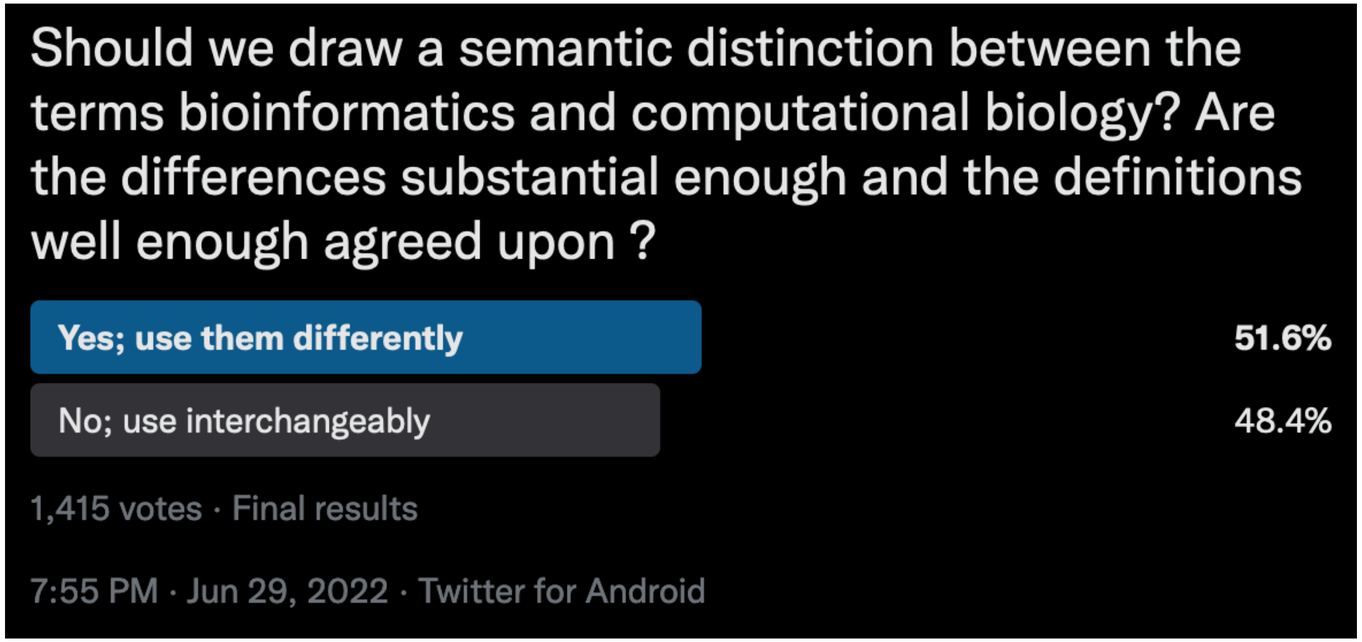
Some of the content is inherently advanced, but it's also *very cool* and *generally useful.* If you're willing to ask questions, I'll do my very best to make sure everyone understands.

Some folks may find some of the projects challenging, that's intended but OK; there is plenty of time to do them, and we give lots of feedback.

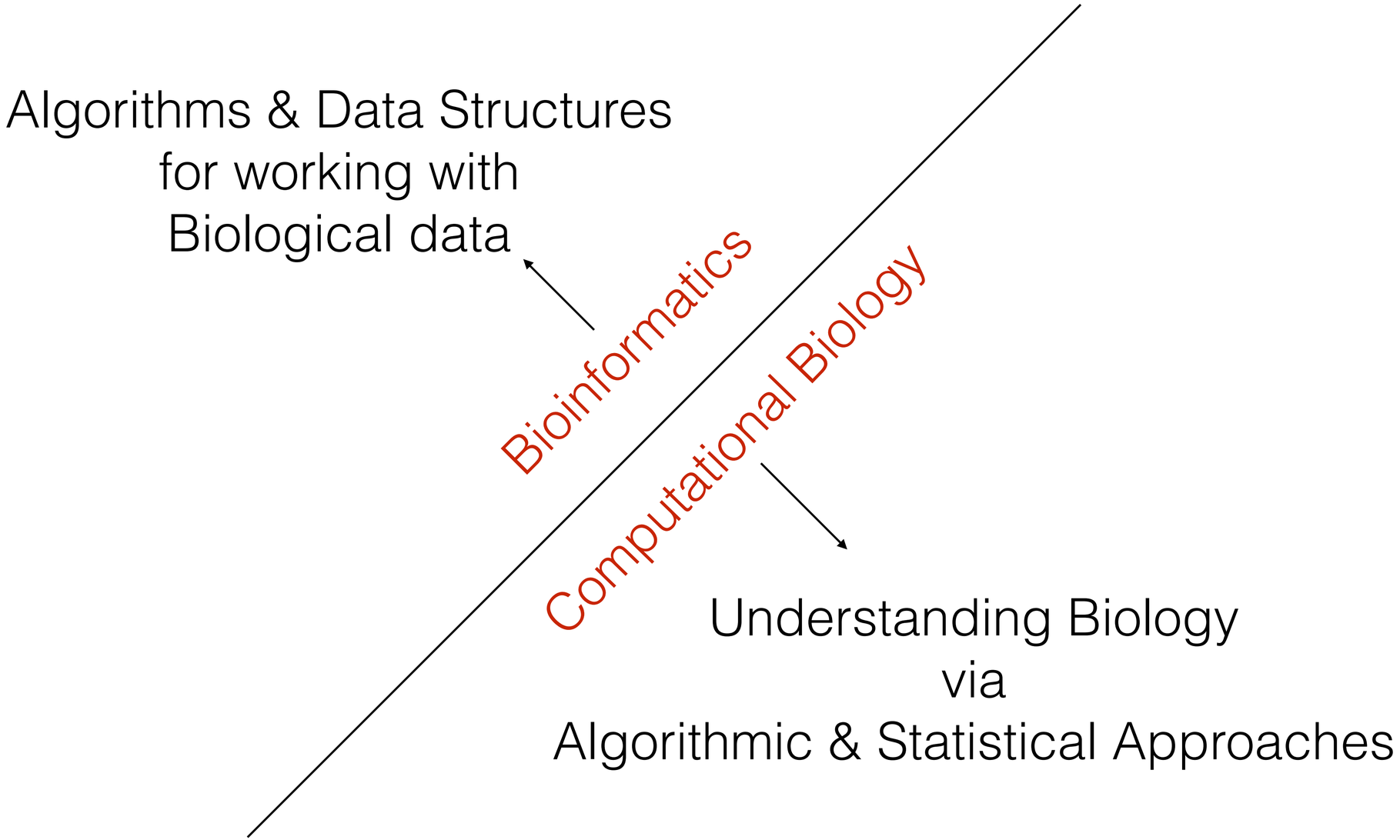
Exams are designed to be somewhat challenging, but fair — we are generous with partial credit & there's almost always a healthy curve!

A rose by any other name





Bioinformatics & Computational Biology



Bioinformatics & Computational Biology

We'll treat this as two sides of the same coin
&
try to ignore this distinction

Why Bioinformatics (genomics)?

Our capabilities for *high-throughput* measurement of Biological data has been transformative

1990 - 2000

Sequencing the first human genome took ~10 years and cost ~$2.7 **billion**

Today

Sequencing a genome costs ~$100 - 1,000✢ (depending on how you count)

~18 Tb per “run” at maximum capacity

Progression of sequencing capacity

$3,000,000,000	**2003** Human Genome Project	
$20,000,000	**2006** 1st individual genome	
$2,000,000	**2007** 1st NGS Genome	
$200,000	**2008** 1st 30x genome	
$10,000	**2010** 1st sub-10K genome	
$1,000	**2014** 1st $1,000 genome	
$100	**2017** 1st $100 genome	

https://blog.dnanexus.com/2017-04-24-dnanexus-celebrates-national-dna-day/

Tons of Data, but we need Knowledge

We'll discuss a bit about how sequencing works soon. But the hallmark *limitations* are:

- Short "reads" (75 — 250) characters when the texts we're interested in are 1,000s to 1,000,000,000s of characters long.

- Imperfect "reads" — results in infrequent but considerable "errors"; modifying, inserting or deleting one or more characters in the "read"

- Biased "reads" — as a result of the underlying chemistry & physics, sampling is not perfectly uniform and random. Biases are not always known.

A cartoon view of sequencing

Orig. genome: GACGATGAAGCCAGCTCGGCTGAACACGATTCCAACGTCTAACGCTCGGTAGGTCTGGCAGGTAGCCGGGCGGAA

Many copies:
GACGATGAAGCCAGCTCGGCTGAACACGATTCCAACGTCTAACGCTCGGTAGGTCTGGCAGGTAGCCGGGCGGAA
GACGATGAAGCCAGCTCGGCTGAACACGATTCCAACGTCTAACGCTCGGTAGGTCTGGCAGGTAGCCGGGCGGAA
GACGATGAAGCCAGCTCGGCTGAACACGATTCCAACGTCTAACGCTCGGTAGGTCTGGCAGGTAGCCGGGCGGAA
…
GACGATGAAGCCAGCTCGGCTGAACACGATTCCAACGTCTAACGCTCGGTAGGTCTGGCAGGTAGCCGGGCGGAA

"random" fragmentation into short sequences:

ATGAAGCCAGC
TCTAACGCTCG
GATTCCAACGT
TAACGCTCGGTAGCTCTGGC

TTCCAACGTCTAA
AACGCTCGGTAGGTC
CGATGAACCCAGCT

CCAACGTCTAAC
CGTCTAACGC
ATTCCAACGTCTAAC

GTCTAACGCTCGGTAGGTC
TGAAGCCAGCTCG
TAAGTCTGGCAGG
TCTGGCAGGGAG

ACGATGAAGCCAGCTCG
CGTCTCACGC
GACCATGAAGCCAGCTCGG
TCTGGCAGGT

CTGATCACGATTCCAACGT
CTGAACACGATTCCA
TTCCAACGTCTAGCGCTTGG
GCTCGGCTGAACACG

sequencing "read"

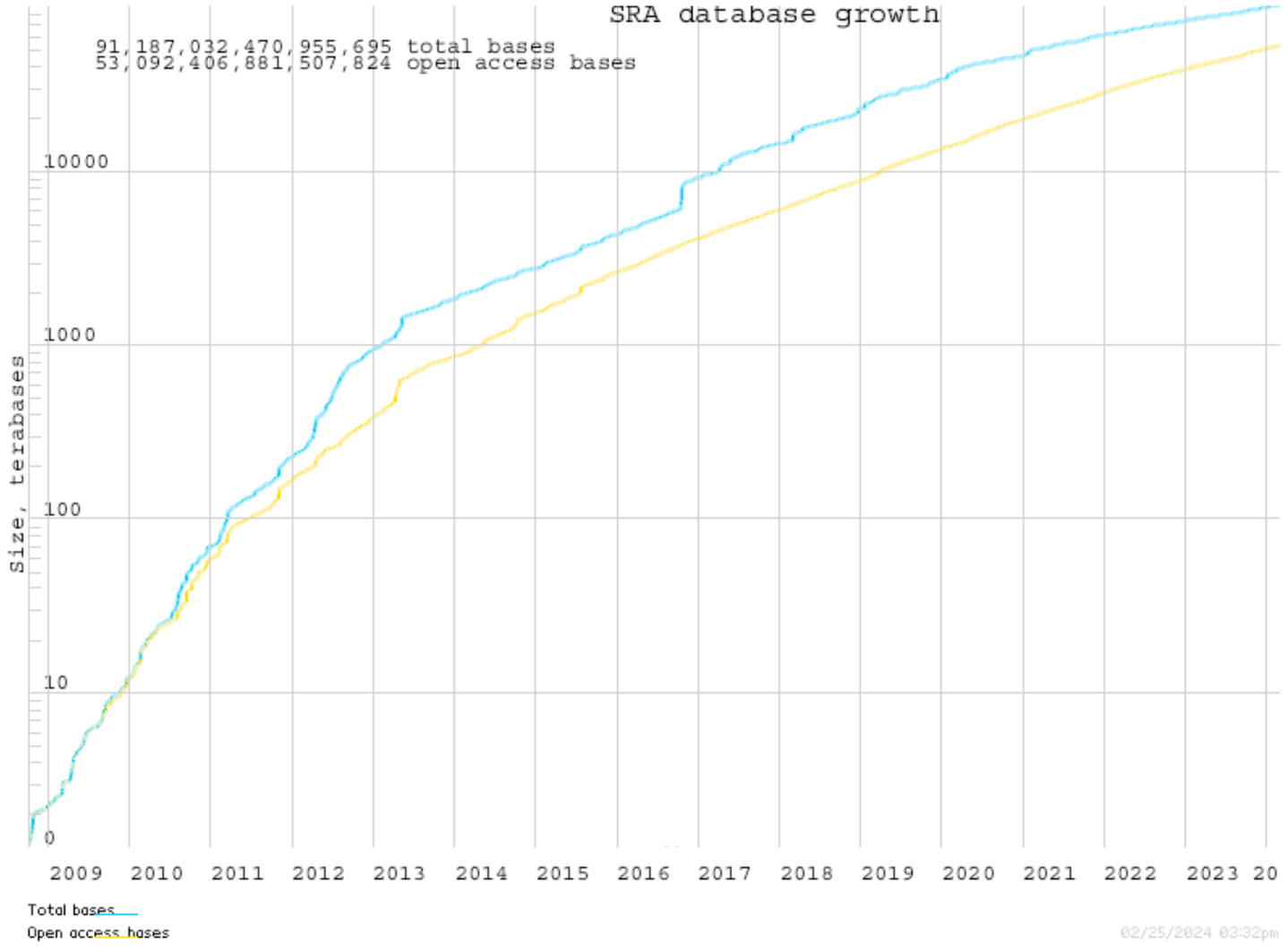
AAGCCAGCTCGGCTGAACA
GCTCGGCTGAACACTATT
TAGGTCTGGCA

TAACGCTCGGTAGGTCTGGC
TGTACACGATTCCGA
CGGCTGAACACGATTC
ACGATTCCAACGTCTAACG
CTCGGAAGGTCTGGCAGG

The FASTA format (relevant for Project 0)

despite these limitations, scientists have used sequencing at a breakneck pace

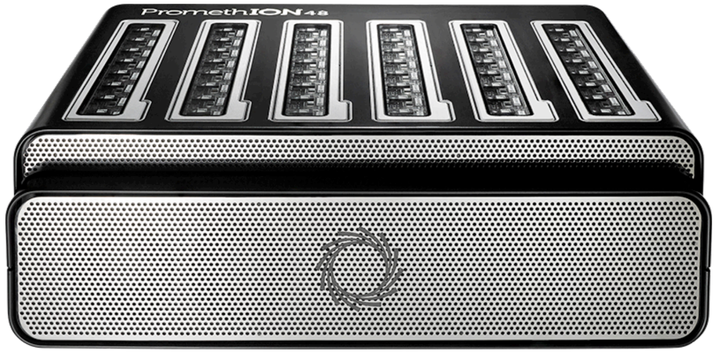
Growth of the Sequence Read Archive (SRA)



data from: http://www.ncbi.nlm.nih.gov/Traces/sra/

Like all technologies, biotech marches forward

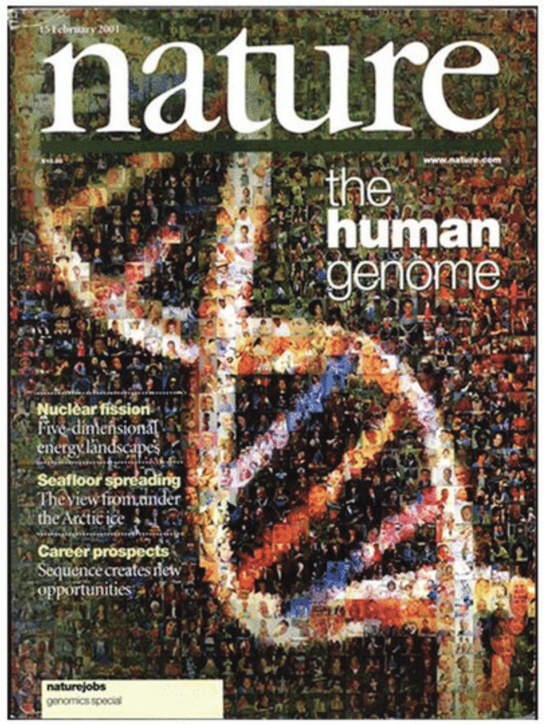
A lot of our focus will be on short-read technologies, but recent improvements in long-read technologies have been transformative:

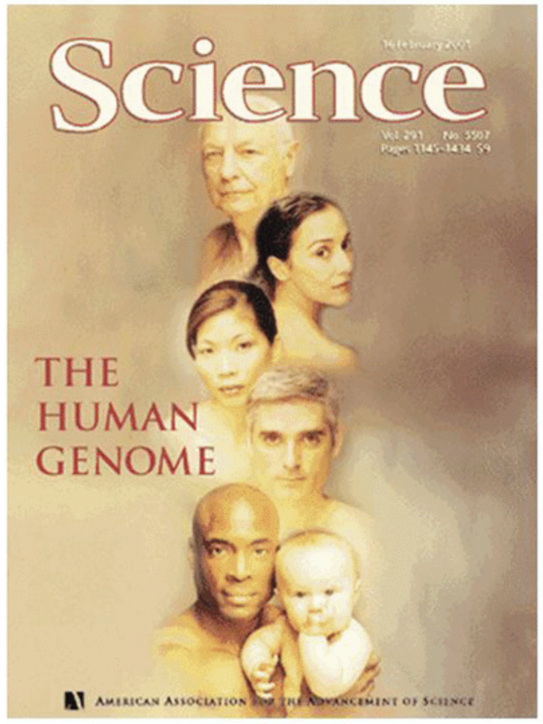


- Much longer reads (10-25kb for PacBio HiFi, up to ~1MB for ONT)
- Many fewer independent samples (long reads, but fewer of them)
- ONT error rate higher than “short reads” 1-5%, PacBio can match Illumina error rates but is *very* expensive

Actually completing the human genome

2001

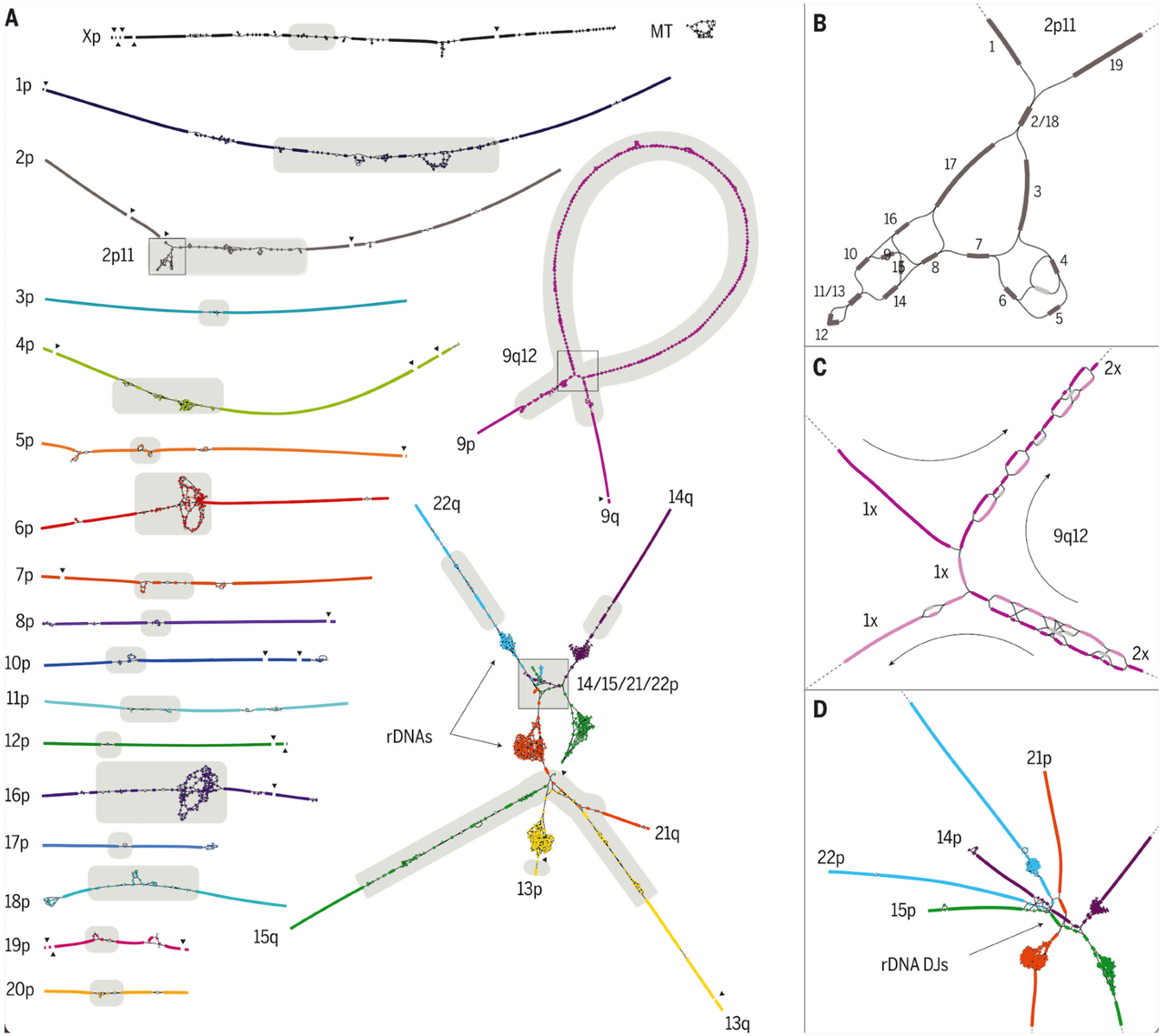




algorithmic
&
biotech
progress

2022

Assembly of the human genome



Nurk, Sergey, et al. "The complete sequence of a human genome." *Science* 376.6588 (2022): 44-53.

Answer questions “in the large”

What is the genome of the terrapin? (genomics)

Which genes are expressed in healthy vs. diseased tissue? (transcriptomics)

How do environment changes affect the microbial ecosystem of the Chesapeake bay? (metagenomics)

How do genome changes lead to changes & diversity in a population? (population genetics/genomics)

How related are two species if we look at their whole genomes? (phylogenetics / phylogenomics)

Some Computational Challenges

Answering questions on such a scale becomes a *fundamentally* computational endeavor:

Assembly — Find a likely "super string" that parsimoniously explains 200M short sub-strings (string processing, graph theory)

Alignment — Find an *approximate* match for 50M short string in a 5GB corpus of text (string processing, data structure & algorithm design)

Expression / Abundance Estimation — Find the most probable mixture of genes / microbes that explain the results of a sequencing experiment (statistics & ML)

Phylogenomics — Given a set of related gene sequences, and an assumed model of sequence evolution, determine how these sequences are related to each other (statistics & ML)

Establishing a basic lexicon

This course will focus on algorithms and data structures (with a little bit of probability & statistics), but the ***problems*** we will explore derive directly from biological questions.

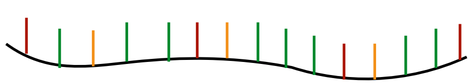
In order to motivate our Computer Science, we will need a basic understanding of the Molecular Biology in which the problems are phrased.

“Flow” of information in the cell

DNA

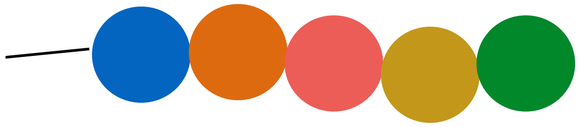
RNA Polymerase
(transcription)

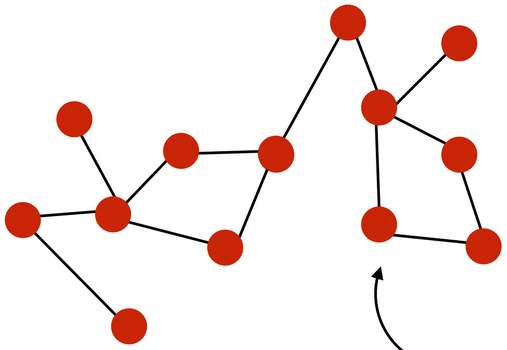
RNA



Ribosomes
(translation)

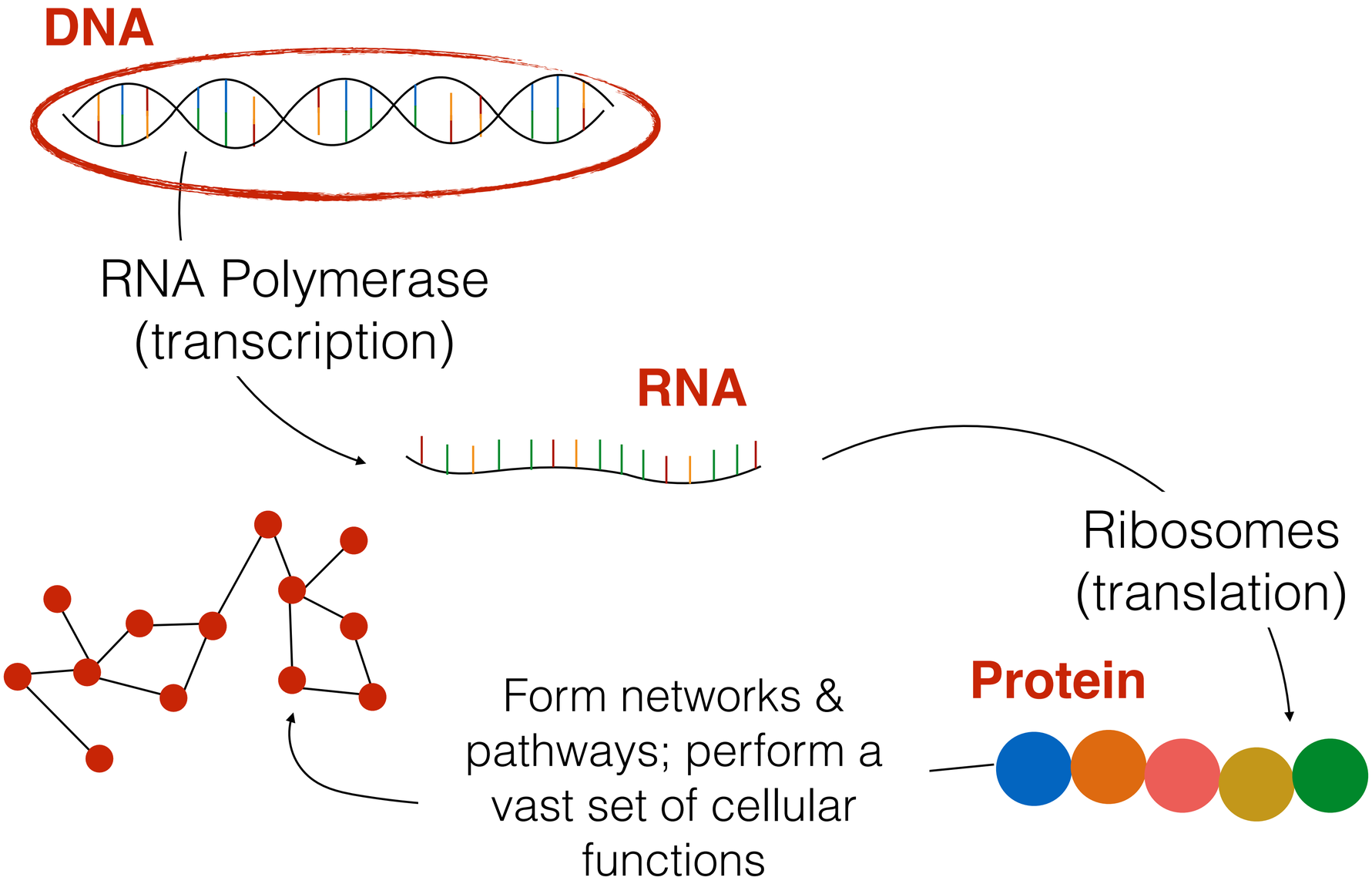
Protein



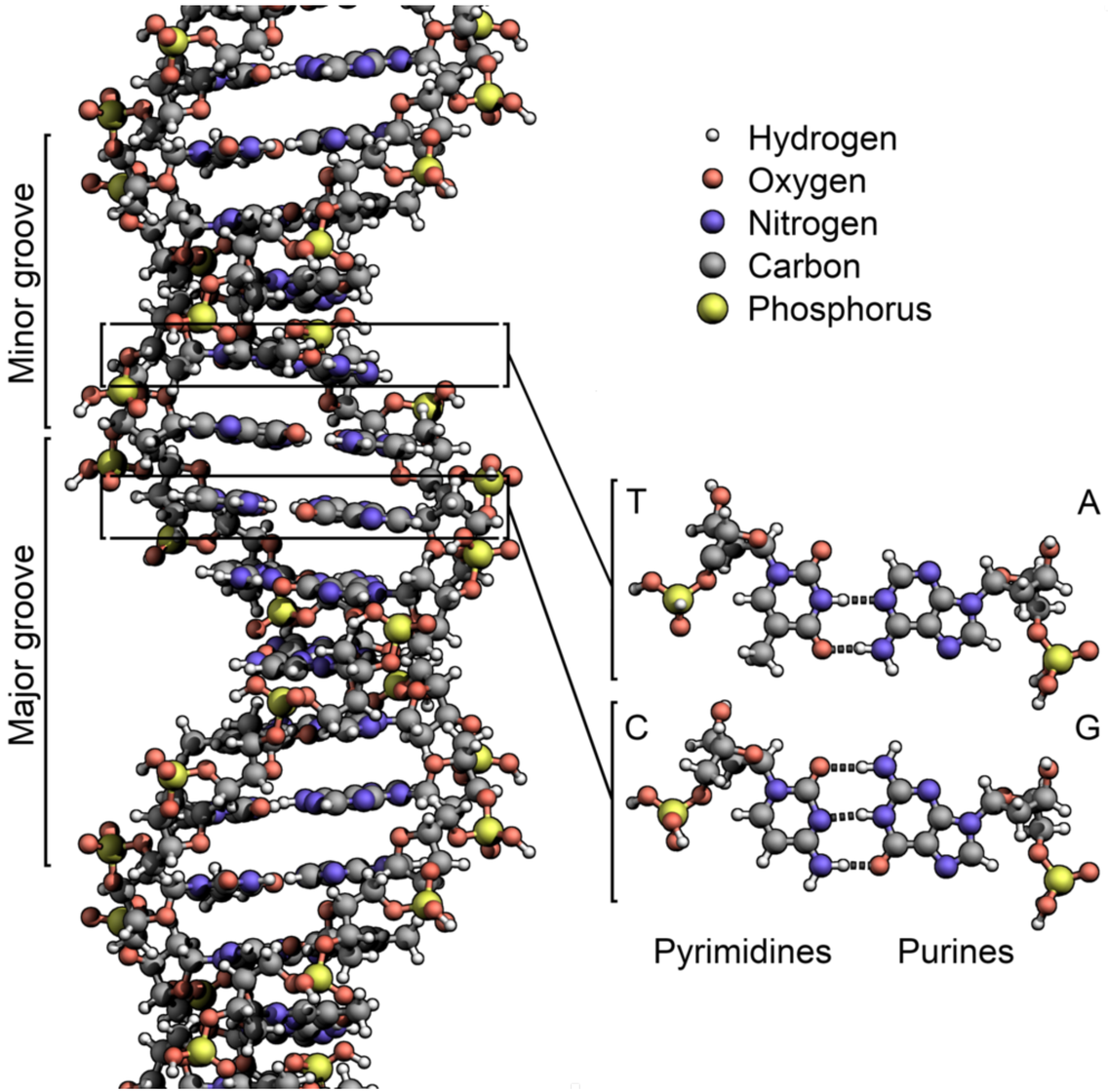


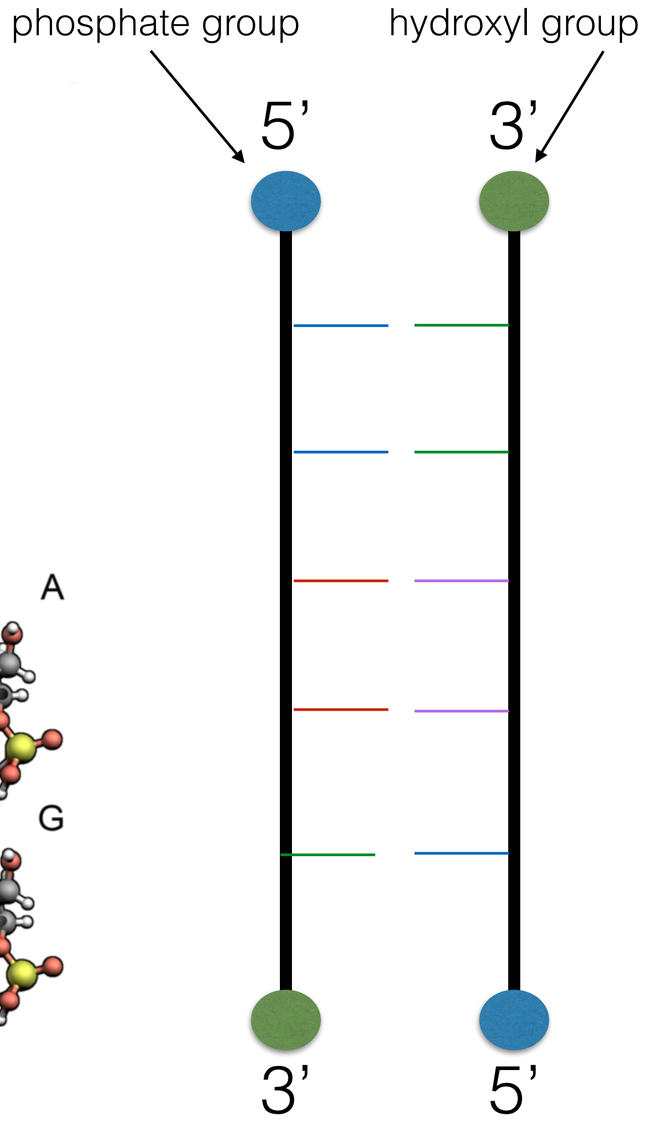
Form networks &
pathways; perform a
vast set of cellular
functions

“Flow” of information in the cell

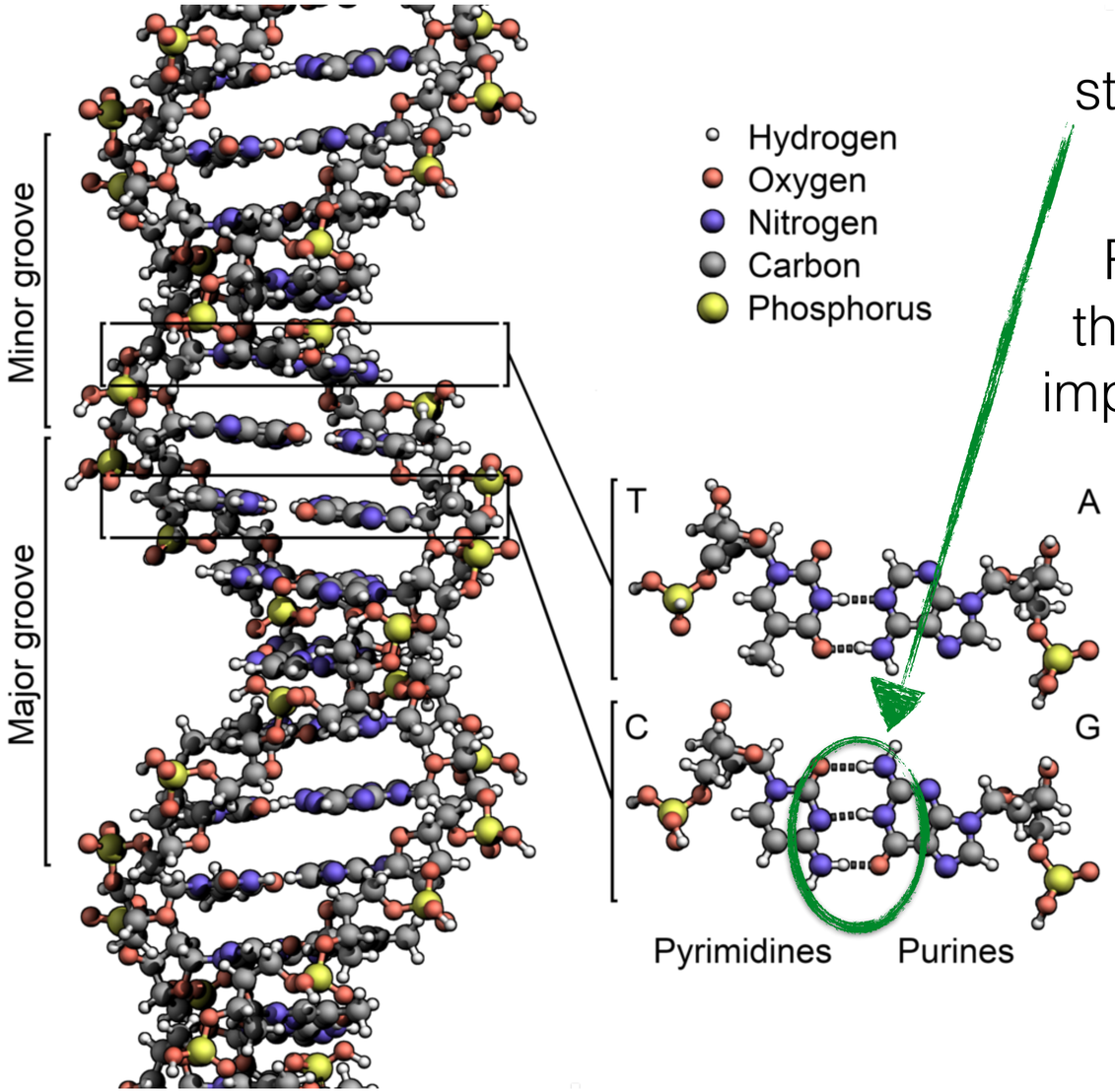


DNA (the genome)





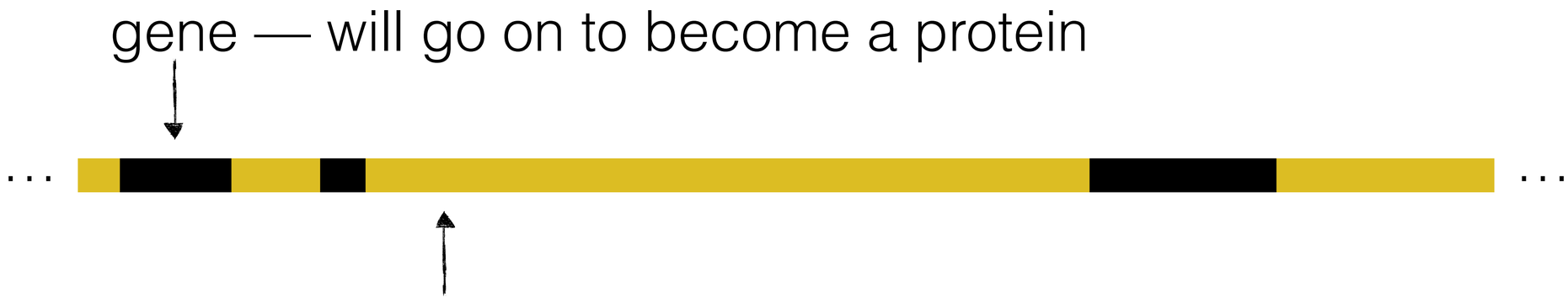
DNA (the genome)



G-C pairing generally stronger than A-T pairing

Ratio of G+C bases — the “GC content” — is an important sequence feature

DNA (the genome)



In humans, most DNA is “non-coding” ~98%

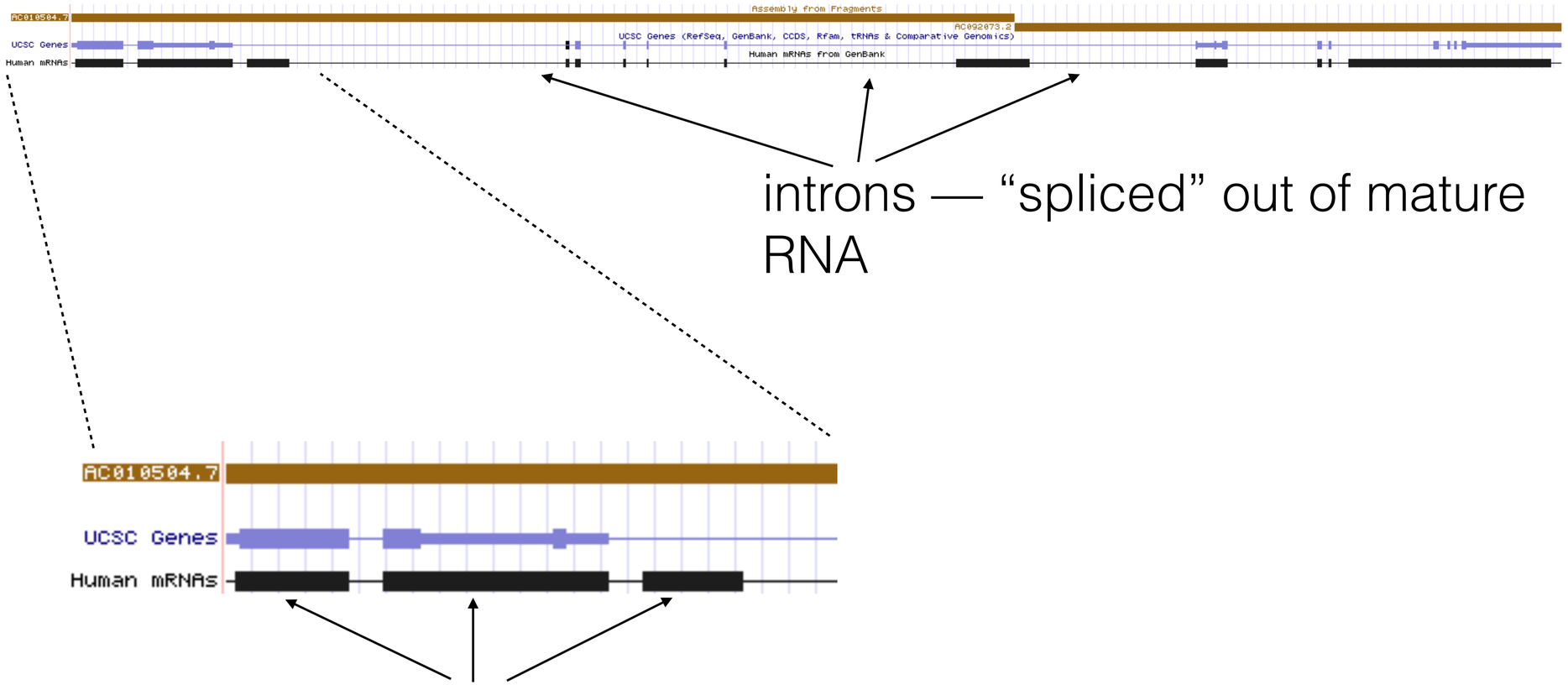
In typical bacterial genome, only small fraction — ~2% — of DNA is “non-coding”

Sometimes referred to as “junk” DNA — much is not, in any way, “junk”

DNA (the genome)

In **prokaryotes**, genes are typically contiguous DNA segment

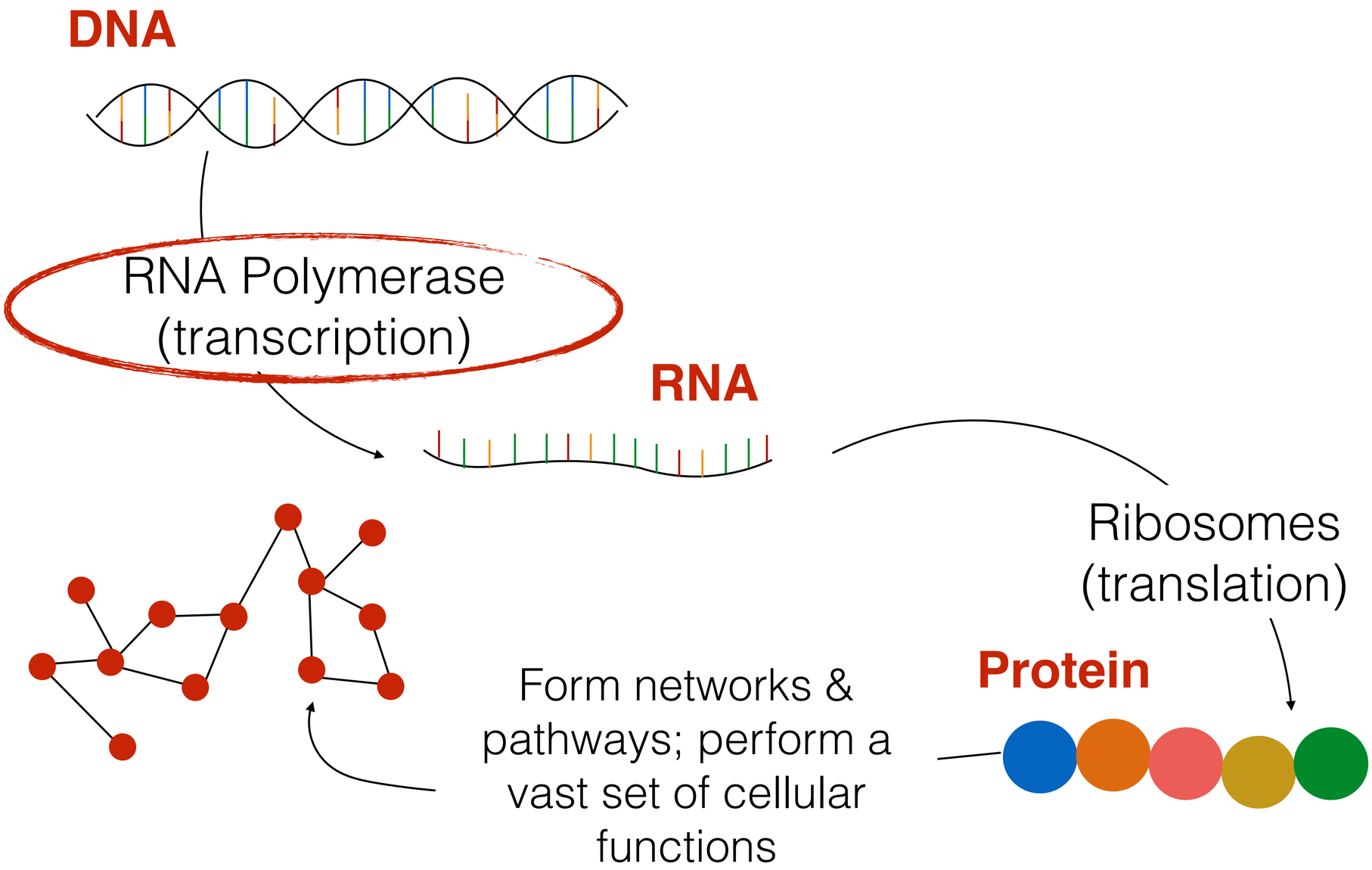
In **eukaryotes**, genes can have complex structure



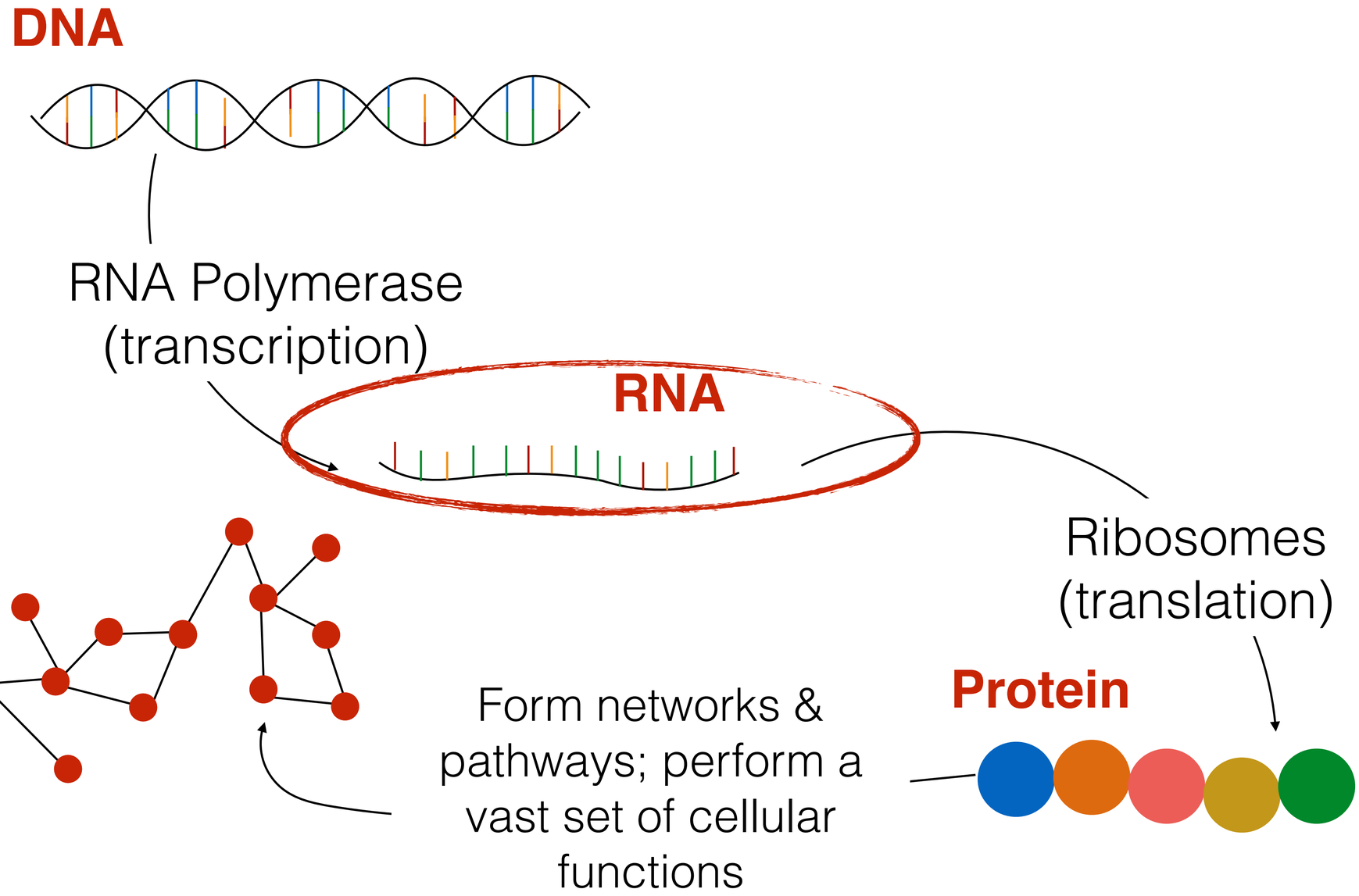
introns — “spliced” out of mature RNA

exons — appear in the *mature* RNA transcript

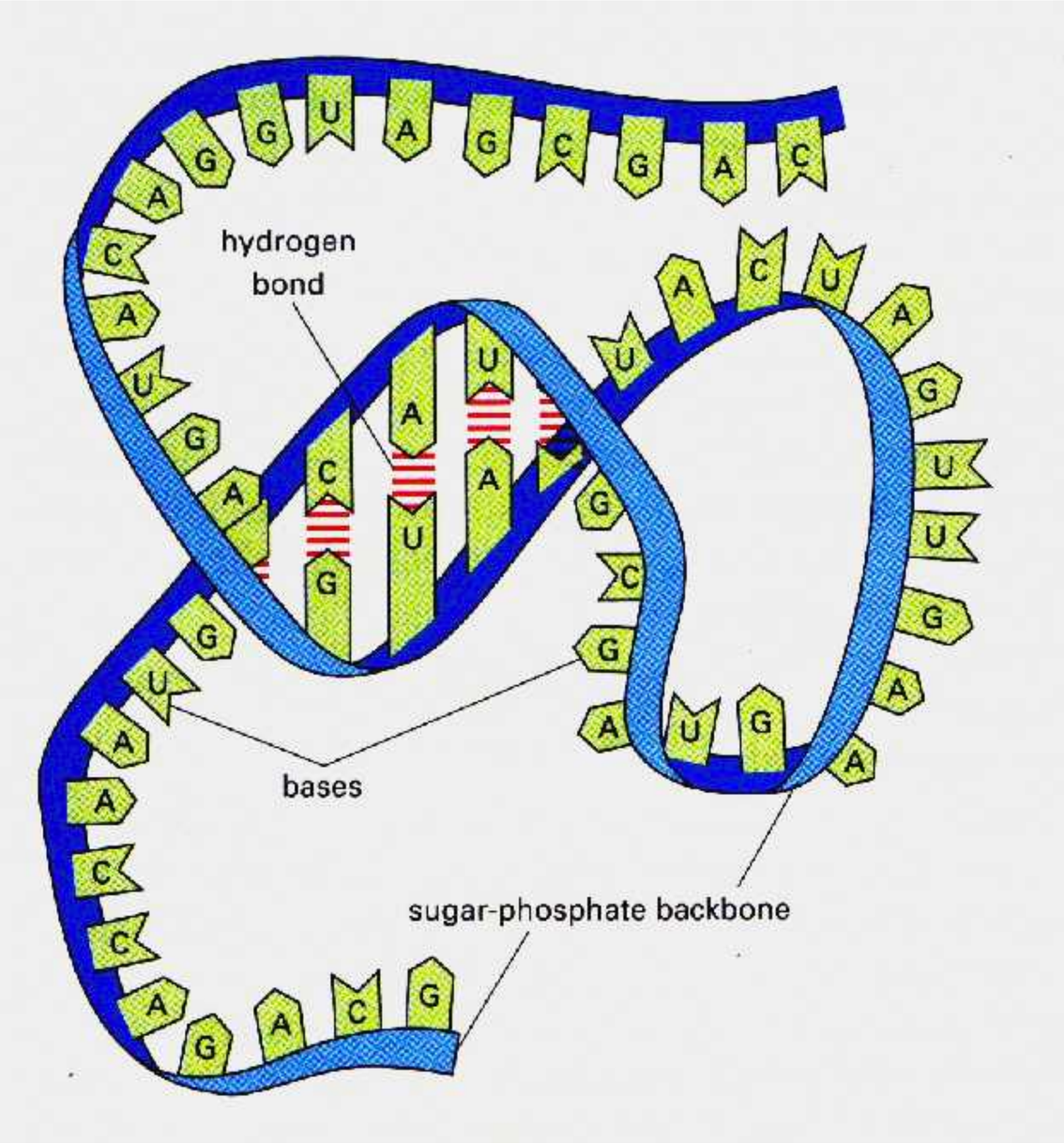
“Flow” of information in the cell



“Flow” of information in the cell



RNA



Less regular structure than DNA

Generally a single-stranded molecule

Secondary & tertiary structure can affect function

Act as transcripts for protein, but <u>also perform important functions themselves</u>

Same “alphabet” as DNA, except thymine replaced by uracil

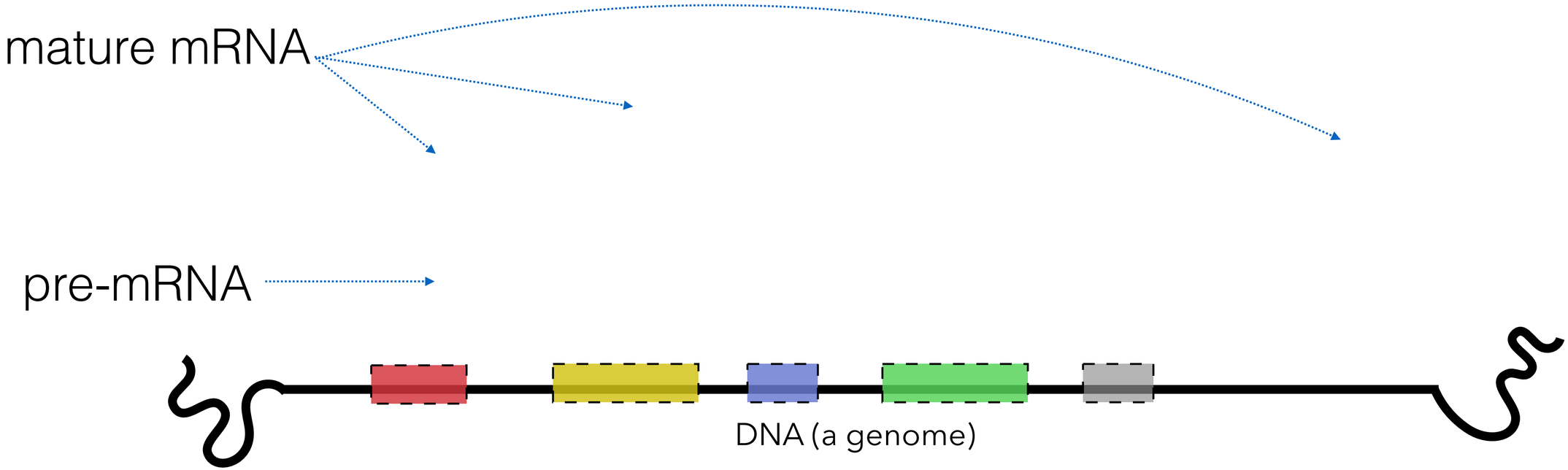
http://tigger.uic.edu/classes/phys/phys461/phys450/ANJUM04/RNA_sstrand.jpg

RNA Splicing

DNA transcribed into pre-mRNA

Some “processing” occurs **capping** & **polyadenylation**

Introns removed from pre-mRNA resulting in mature mRNA

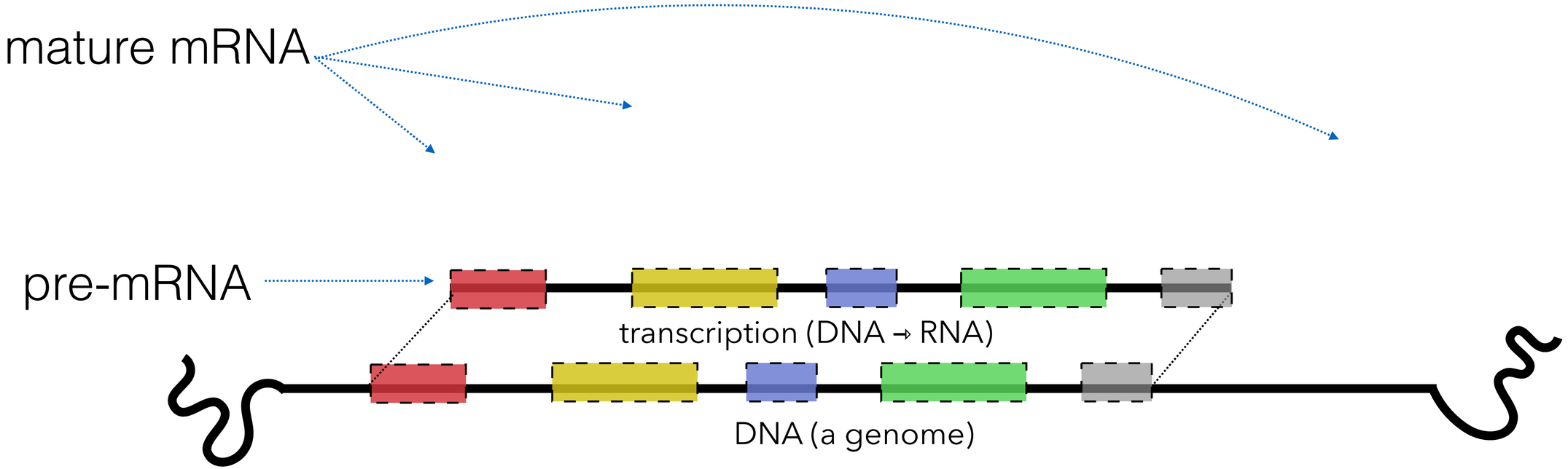


RNA Splicing

DNA transcribed into pre-mRNA

Some “processing” occurs **capping** & **polyadenylation**

Introns removed from pre-mRNA resulting in mature mRNA

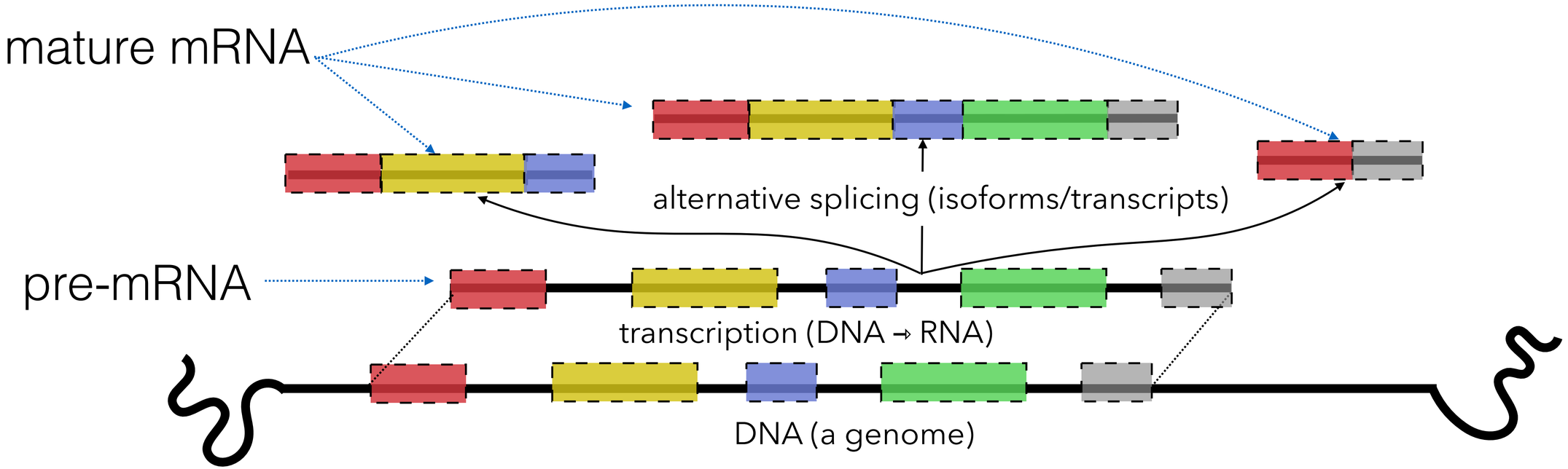


RNA Splicing

DNA transcribed into pre-mRNA

Some “processing” occurs **capping** & **polyadenylation**

Introns removed from pre-mRNA resulting in mature mRNA



“Flow” of information in the cell

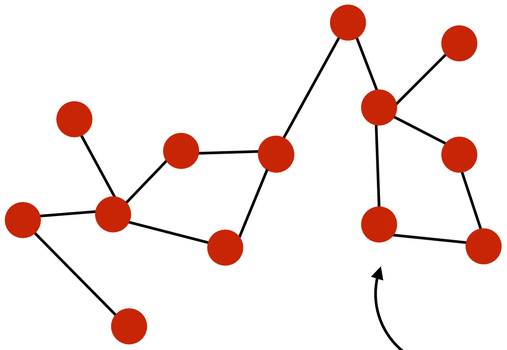
DNA

RNA Polymerase
(transcription)

RNA

Ribosomes
(translation)

Protein



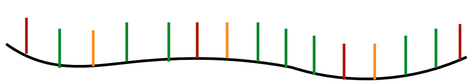
Form networks &
pathways; perform a
vast set of cellular
functions

“Flow” of information in the cell

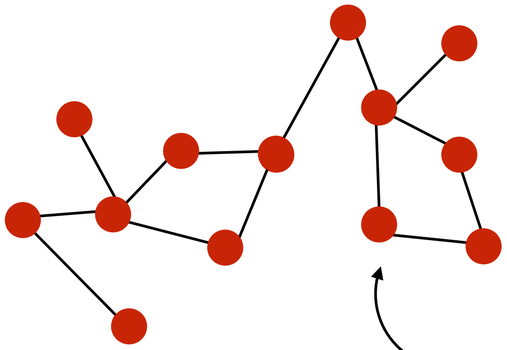
DNA

RNA Polymerase
(transcription)

RNA



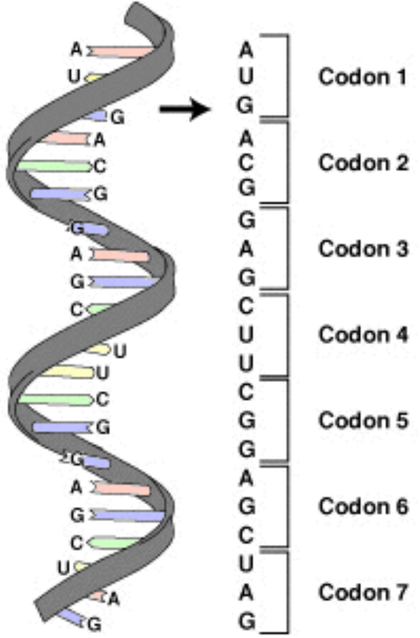
Ribosomes
(translation)



Form networks &
pathways; perform a
vast set of cellular
functions

Protein

Protein



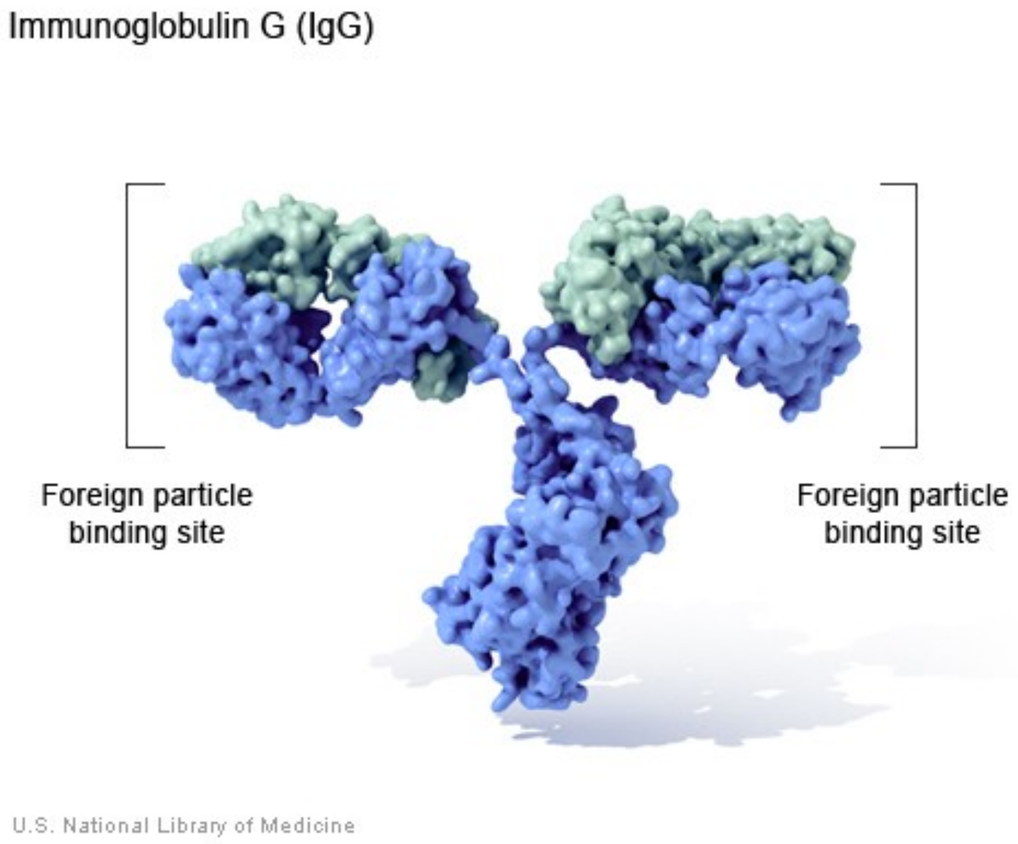
Triplets of mRNA bases (codons) correspond to specific amino acids

This mapping is known as the “genetic code” — an *almost* law of molecular Biology

Inverse table (compressed using IUPAC notation)

Amino acid	Codons	Compressed	Amino acid	Codons	Compressed
Ala/A	GCU, GCC, GCA, GCG	GCN	**Leu/L**	UUA, UUG, CUU, CUC, CUA, CUG	YUR, CUN
Arg/R	CGU, CGC, CGA, CGG, AGA, AGG	CGN, MGR	**Lys/K**	AAA, AAG	AAR
Asn/N	AAU, AAC	AAY	**Met/M**	AUG	
Asp/D	GAU, GAC	GAY	**Phe/F**	UUU, UUC	UUY
Cys/C	UGU, UGC	UGY	**Pro/P**	CCU, CCC, CCA, CCG	CCN
Gln/Q	CAA, CAG	CAR	**Ser/S**	UCU, UCC, UCA, UCG, AGU, AGC	UCN, AGY
Glu/E	GAA, GAG	GAR	**Thr/T**	ACU, ACC, ACA, ACG	ACN
Gly/G	GGU, GGC, GGA, GGG	GGN	**Trp/W**	UGG	
His/H	CAU, CAC	CAY	**Tyr/Y**	UAU, UAC	UAY
Ile/I	AUU, AUC, AUA	AUH	**Val/V**	GUU, GUC, GUA, GUG	GUN
START	AUG		**STOP**	UAA, UGA, UAG	UAR, URA

Protein



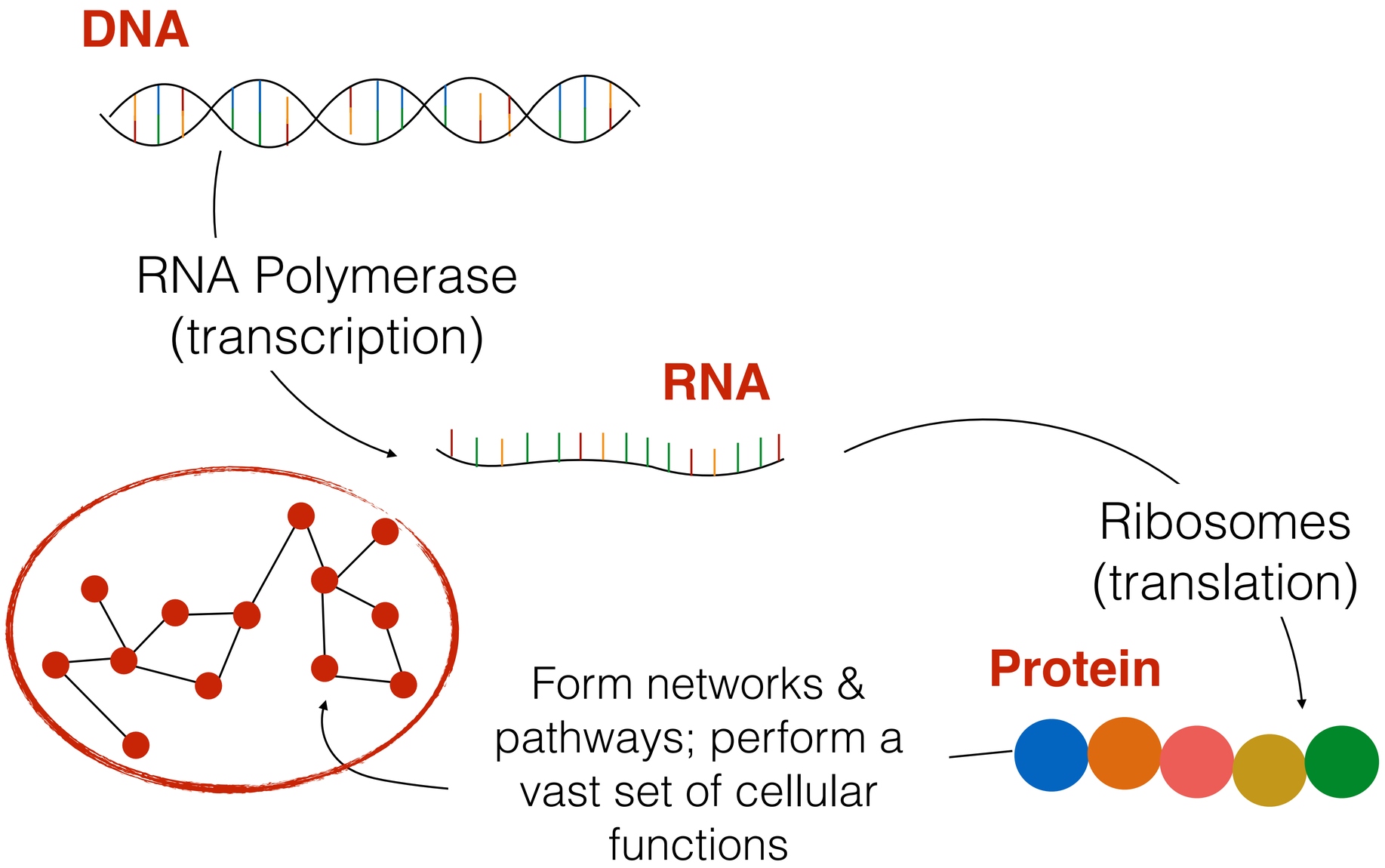
Perform vast majority of intra & extra cellular functions

Can range from a few amino acids to *very* large and complex molecules

Can bind with other proteins to form protein complexes

The shape or *conformation* of a protein is intimately tied to its function. Protein shape, therefore, is strongly conserved through evolution — even more so than sequence. A protein can undergo sequence mutations, but fold into the same or a similar shape and still perform the same function.

“Flow” of information in the cell



One way in which this “central dogma” is violated … retroviruses

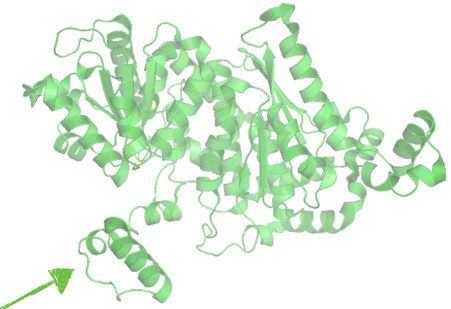
Glycolysis Pathway

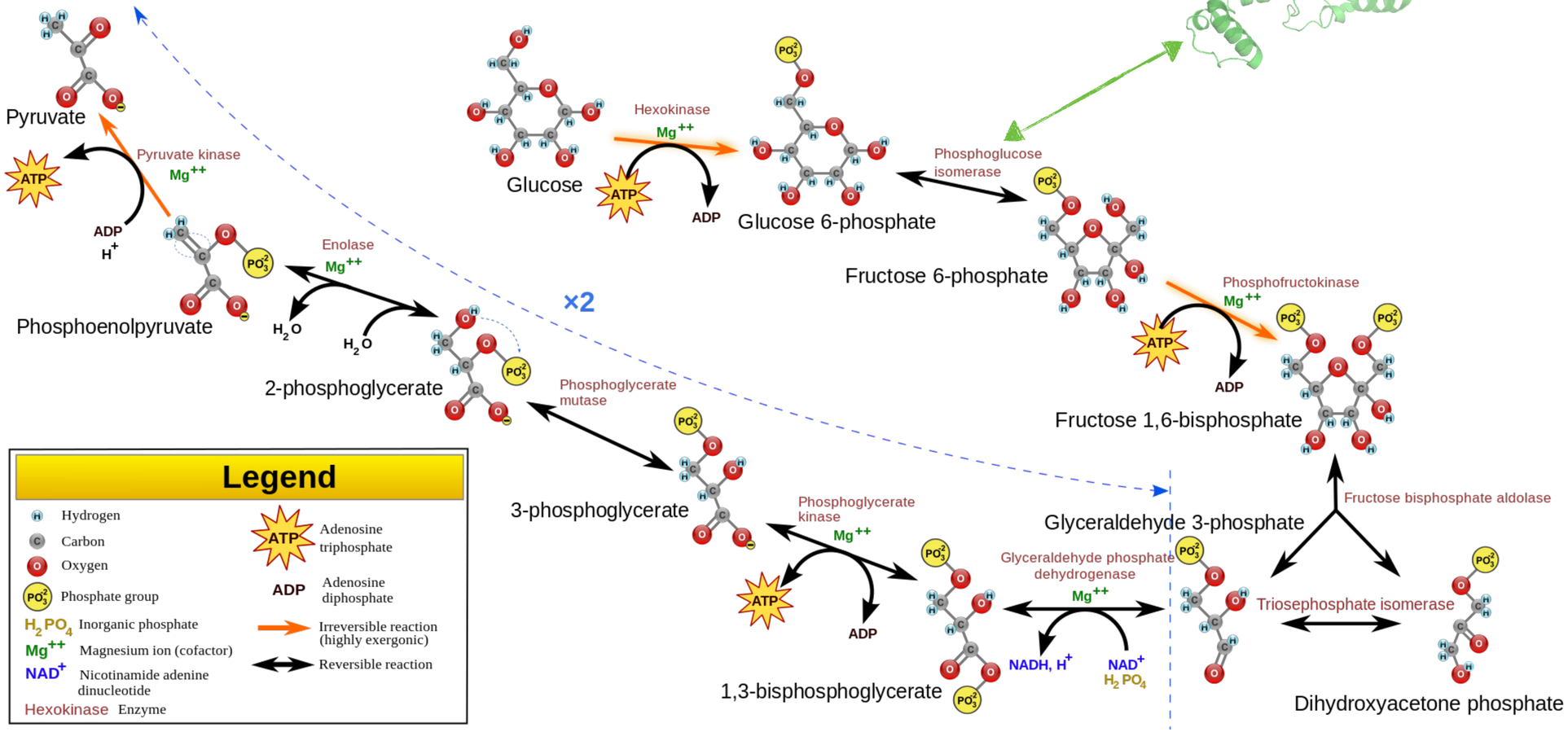
Converts glucose → pyruvate

Generates ATP (“energy currency” of the cell)

this is an **example**, no need to memorize this Bio.

phosphoglucose isomerase





Some Interesting Facts

Organism	Genome size	# of genes (protein coding)
ϕX174 (*E. coli* virus)	~5kb	11
E. coli K-12	~4.6Mb	~4,300
Fruit Fly	~122Mb	~17,000
Human	~3.05Gb	~21,000
Mouse	~2.8Gb	~23,000
P. abies (a spruce tree)	~19.6Gb	~28,000

No strong link between genome size & phenotypic complexity

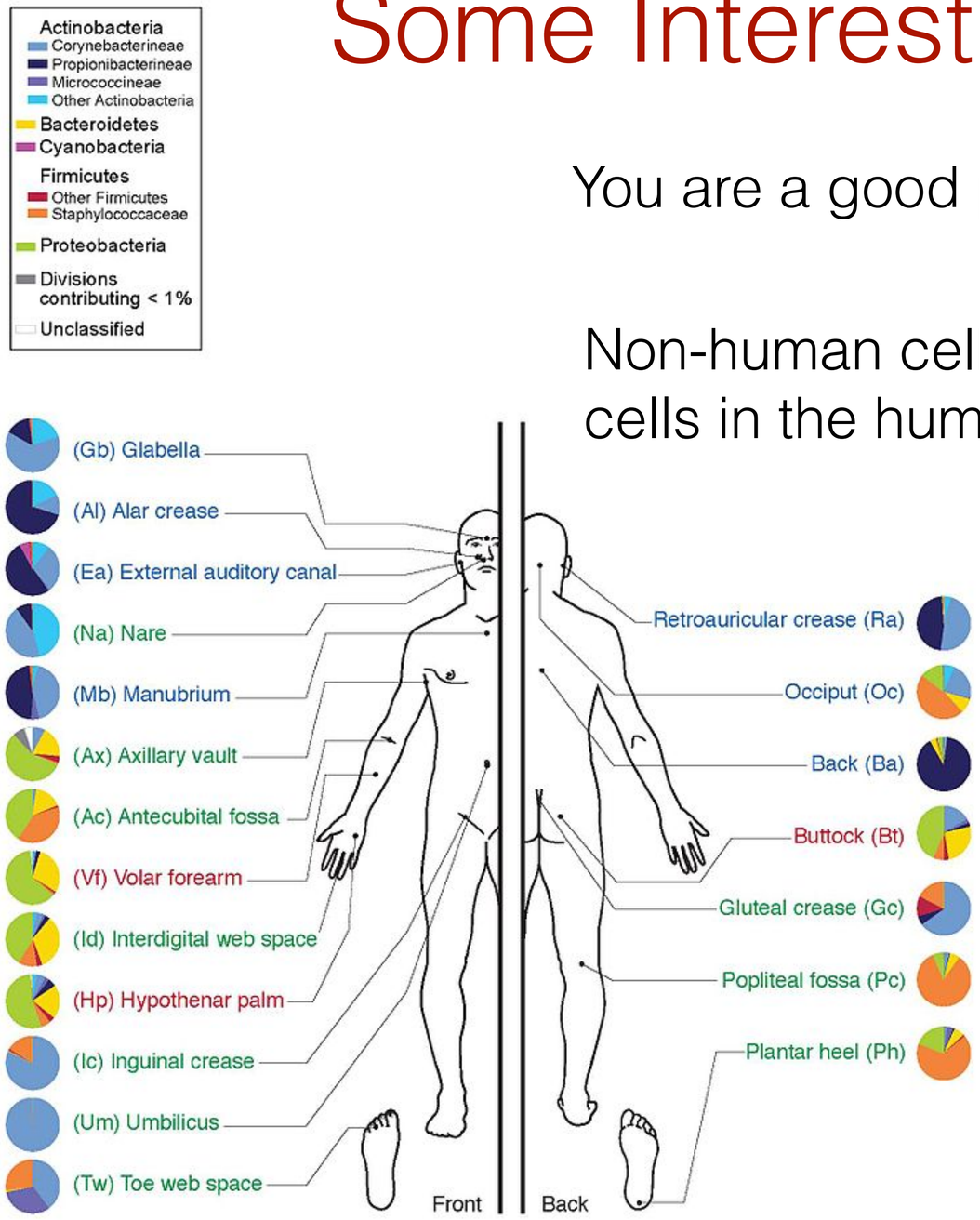
Plants can have **huge** genomes (adapt to environment while stationary!)

http://users.rcn.com/jkimball.ma.ultranet/BiologyPages/G/GenomeSizes.html

Some Interesting Facts

You are a good part non-human cells (e.g. bacteria)

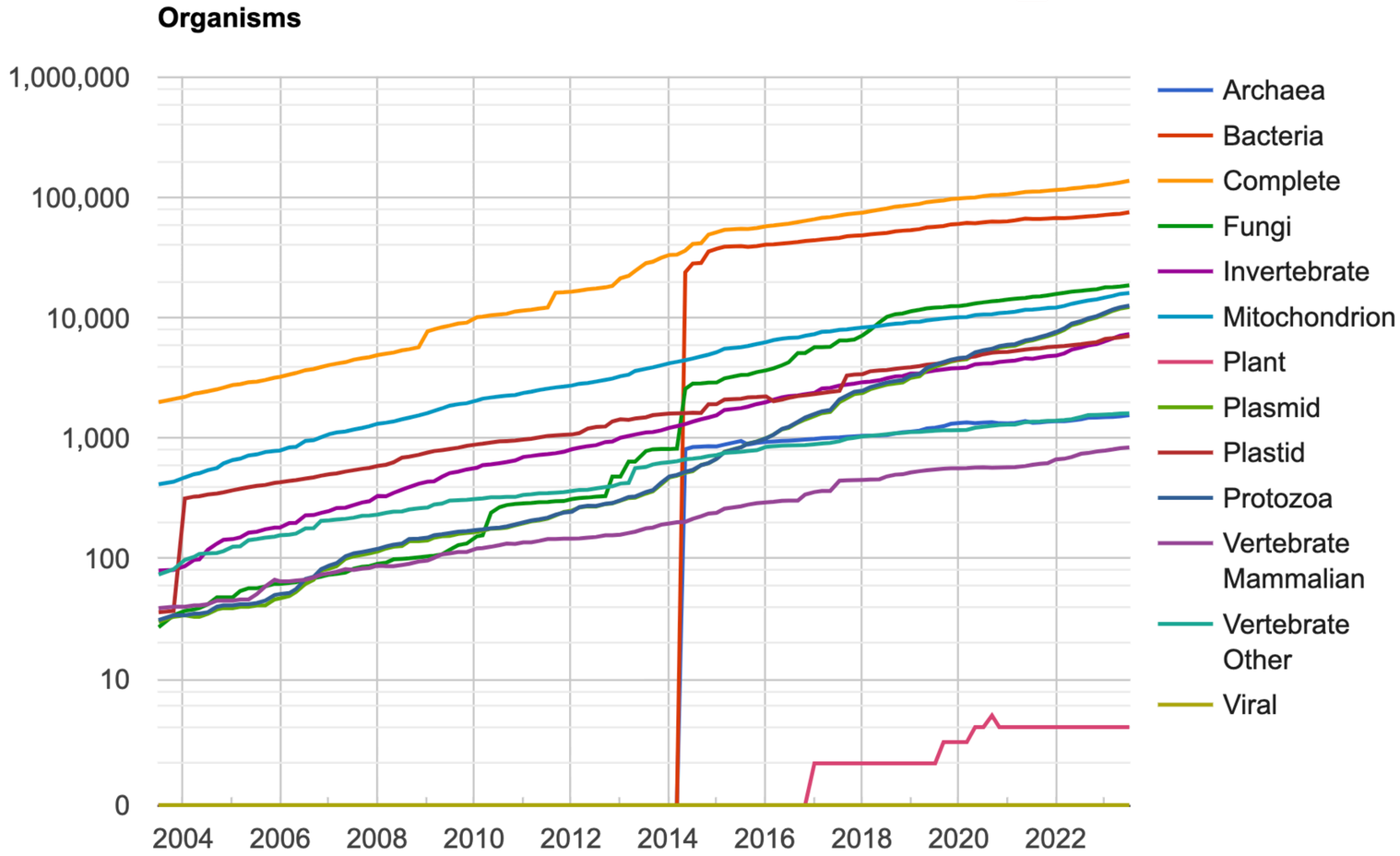
Non-human cells equal or outnumber human cells in the human body



This population of organisms is called the microbiome

en.wikipedia.org : public domain

Some Interesting Facts



. . . Out of 8.7 ± 1.3 Mil*

Vast majority of species unsequenced & *can not be cultivated in a lab* (one of the many motivations for metagenomics)

*Mora, Camilo, et al. "How many species are there on Earth and in the ocean?." PLoS biology 9.8 (2011): e1001127.

https://www.ncbi.nlm.nih.gov/refseq/statistics/