Bioinformatics

CS300

Crash course:

Transcription and Translation

Running Python in Docker or Online

Spring 2021

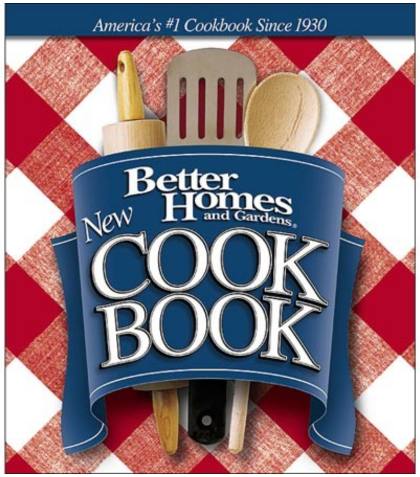
Oliver BONHAM-CARTER

Genes vs Gene Expression

All genes are present in the genome genes only expressed when needed

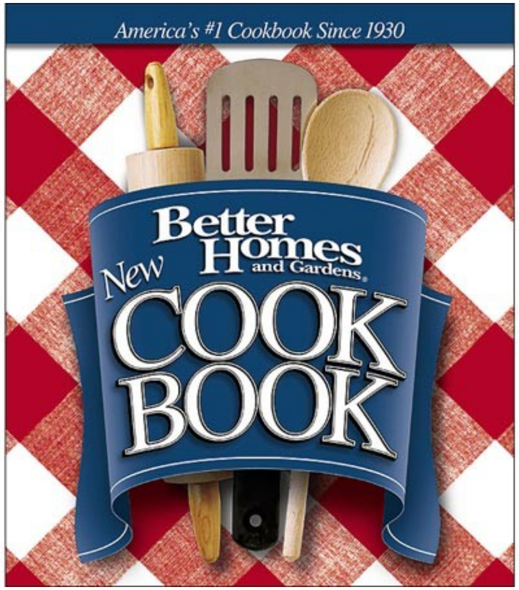
Of the many recipes in the cookbook…

… Only transcribe and translate 4th of July recipes in **July**



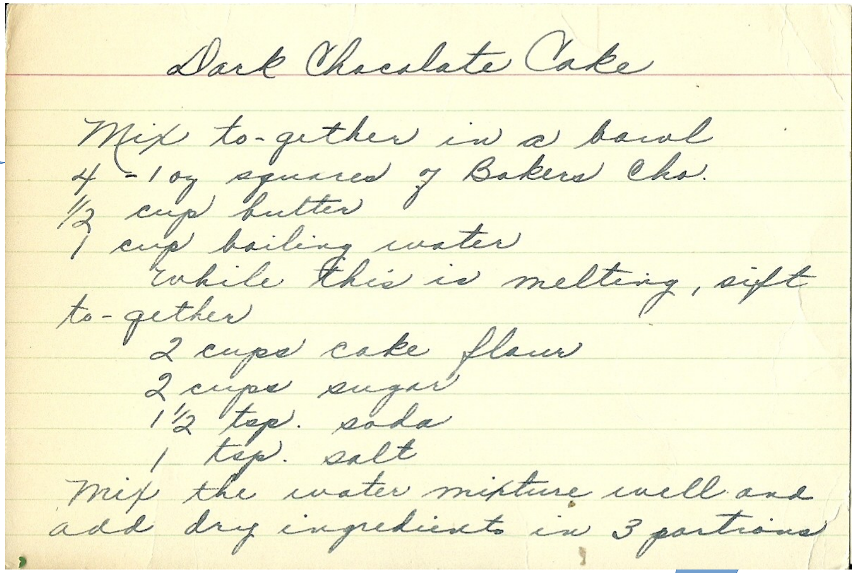
… Only transcribe and translate the Thanksgiving turkey recipe in **November**

Transcription and Translation (Gene Expression)



Transcription

copy a set of ingredients/instructions from a cookbook to create a recipe



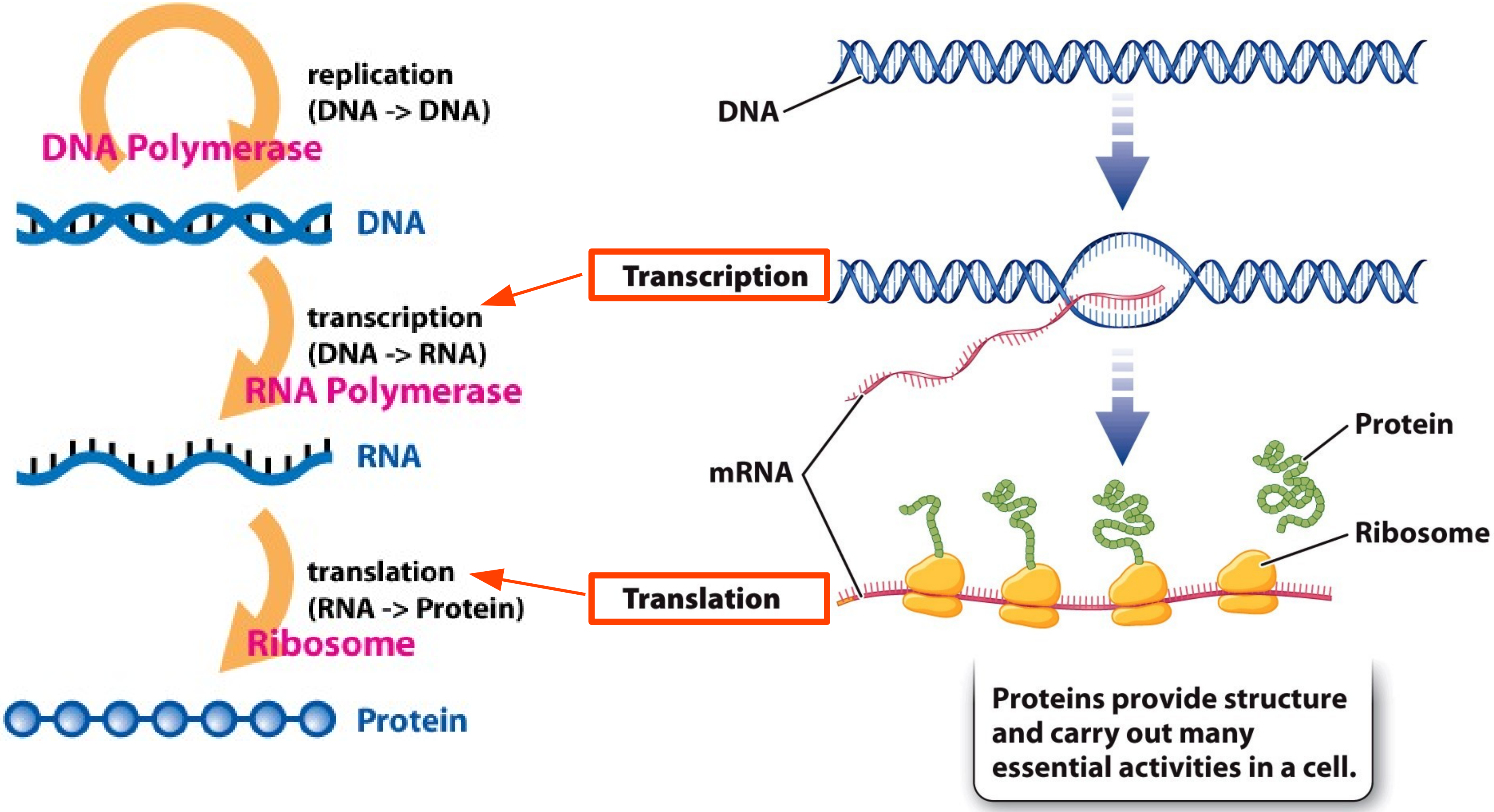
Dark Chocolate Cake

Mix to-gether in a bowl
4 - 1 oz squares of Bakers Cho.
1/2 cup butter
1 cup boiling water
While this is melting, sift
to-gether
2 cups cake flour
2 cups sugar
1 1/2 tsp. soda
1 tsp. salt
Mix the water mixture well and
add dry ingredients in 3 portions

Translation

use the recipe to create a dish

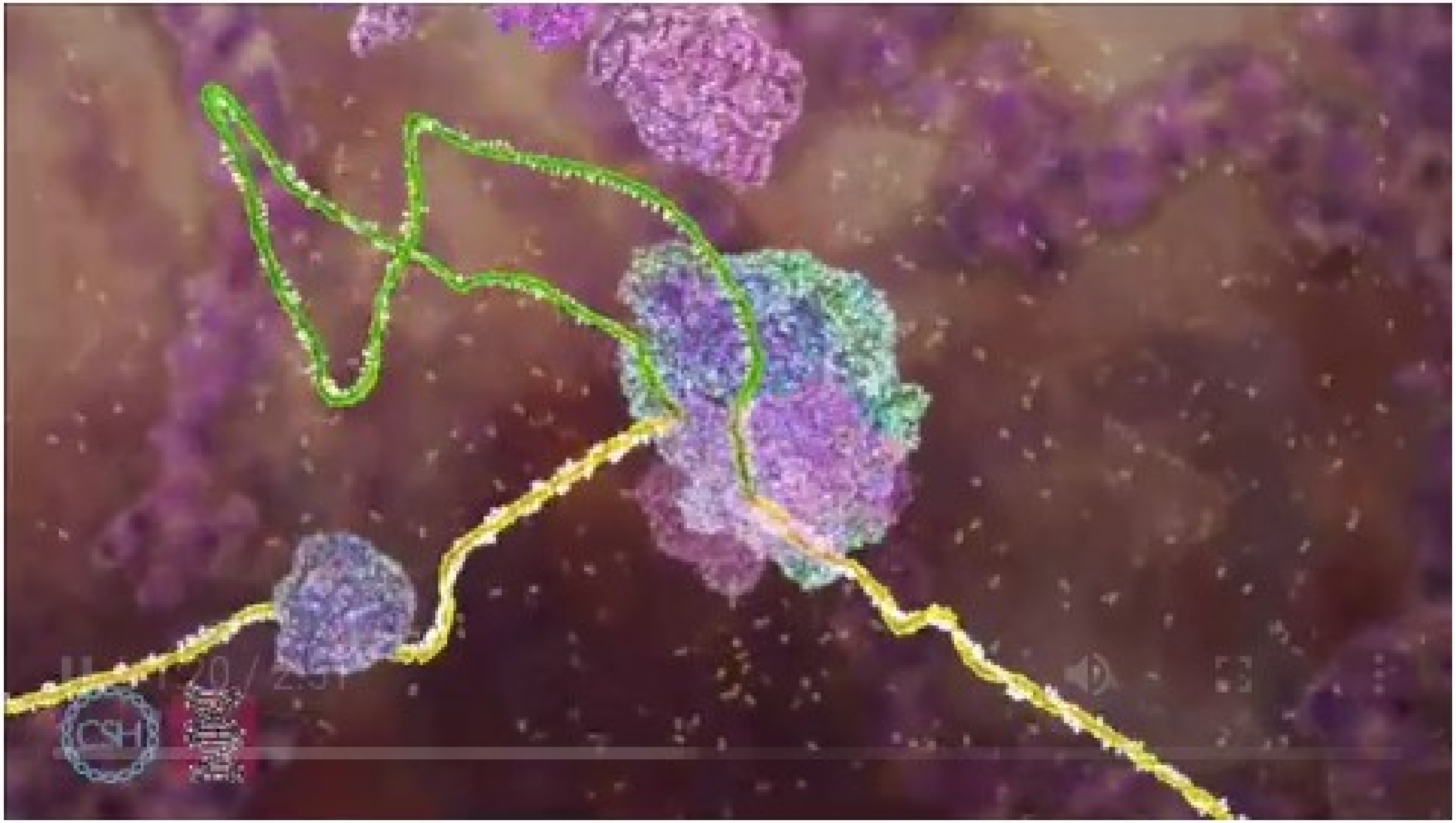
Transcription and Translation



Proteins provide structure and carry out many essential activities in a cell.

Figure 3.3
How Life Works
© 2014 W. H. Freeman and Company

Animation:
DNA → RNA → Protein

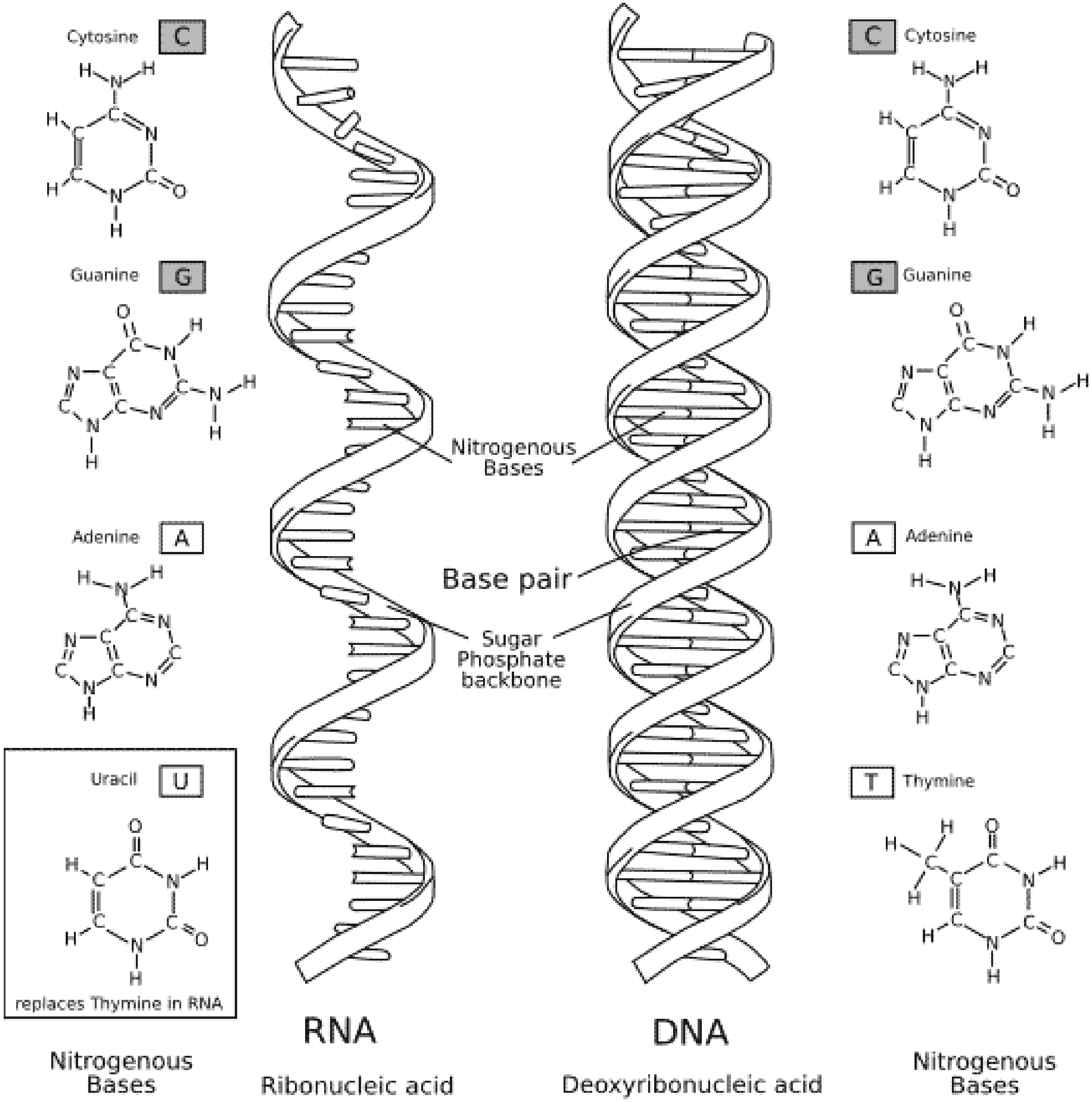


Cold Spring Harbor Laboratory link:
https://dnalc.cshl.edu/view/16933-3D-Animation-of-DNA-to-RNA-to-Protein.html

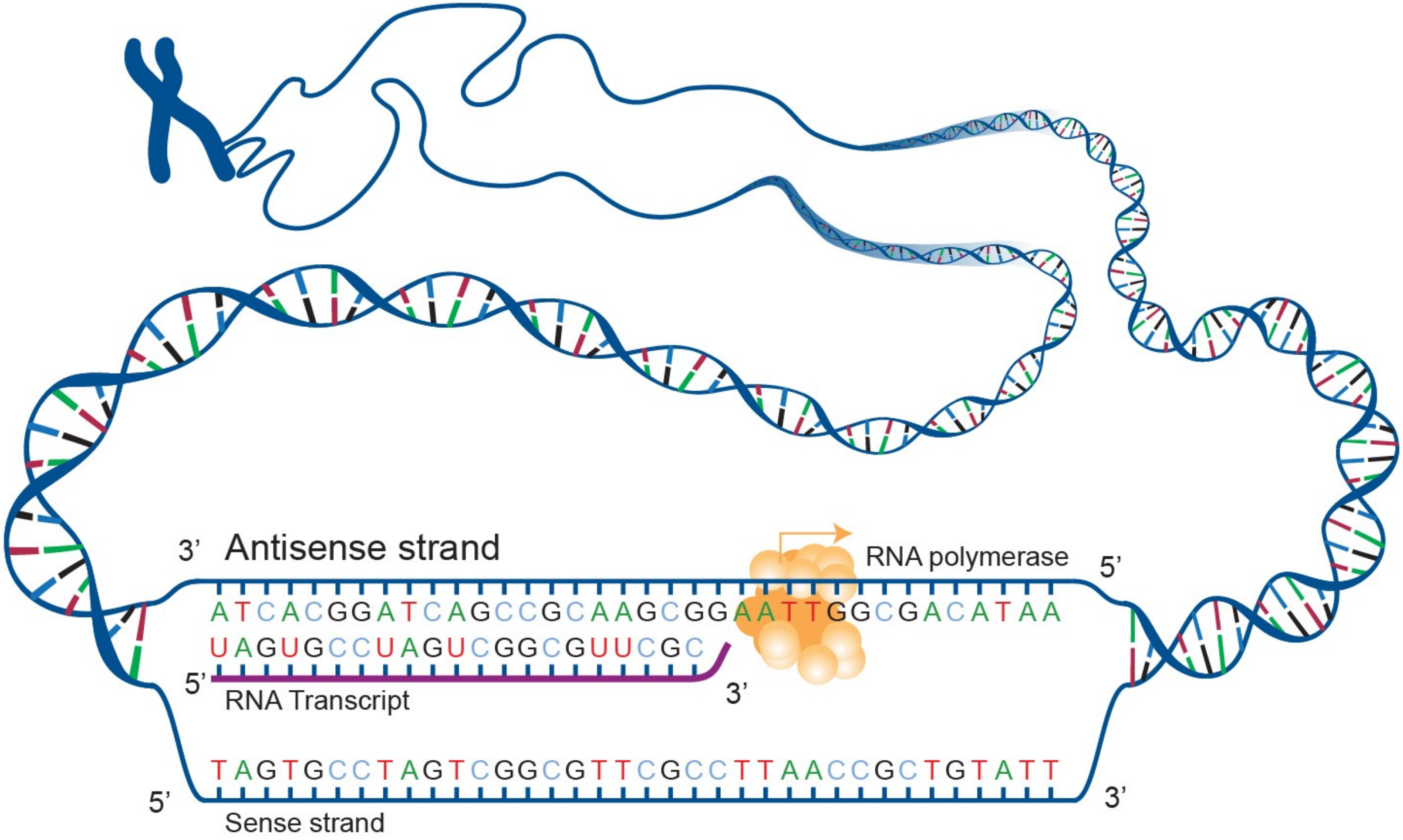
Structure: RNA vs DNA

- RNA – uracil replaces thymine (no Ts in RNA)

- RNA – single stranded (one backbone, no basepairs)



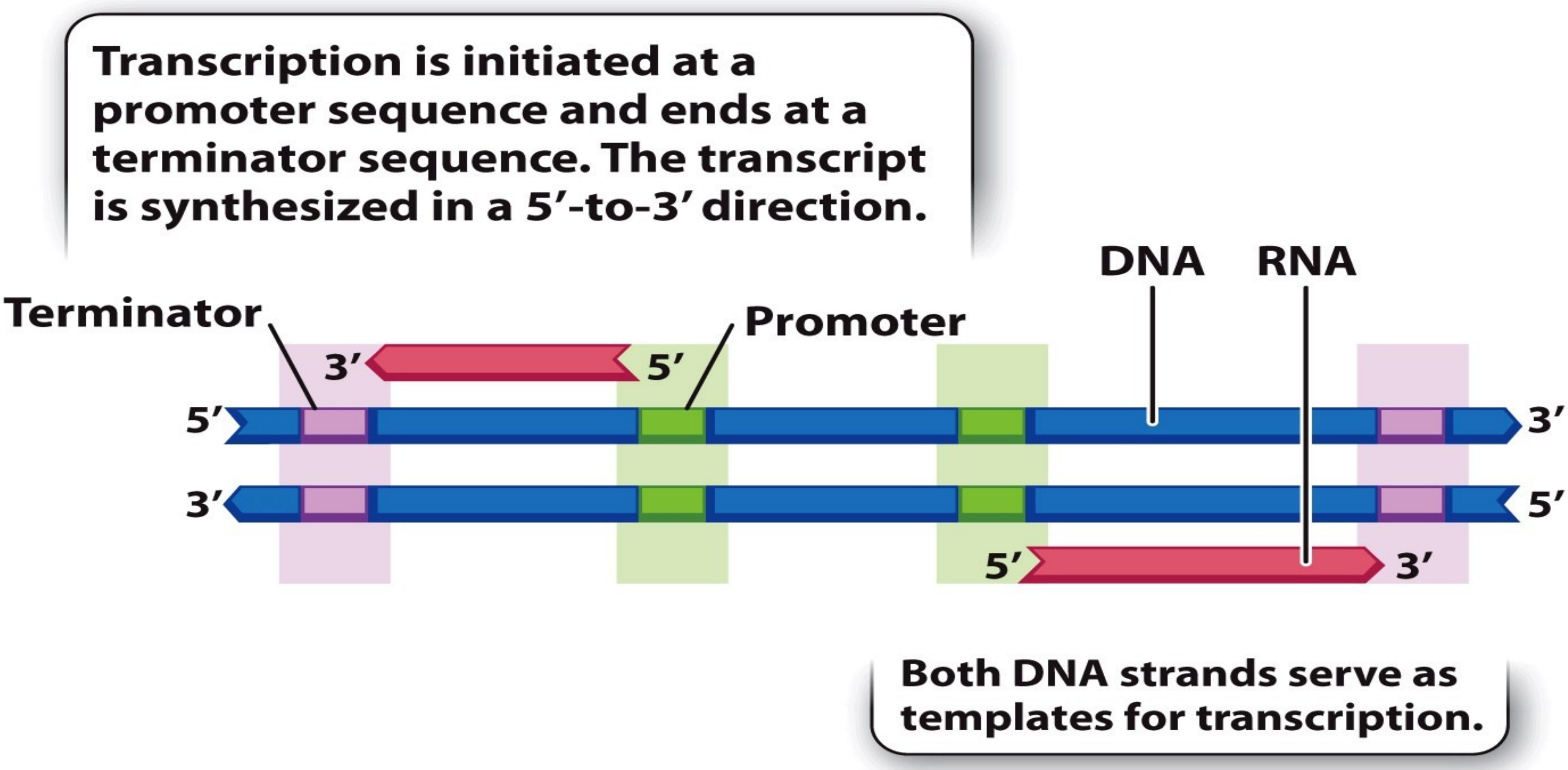
Antisense and Sense Strands of DNA – Relative to the gene being transcribed



Genes for coding proteins are found on “antisense” strands. Their complements are found on the “sense” strand.

Beginnings and Endings of Genes
Promoters and *Terminators*

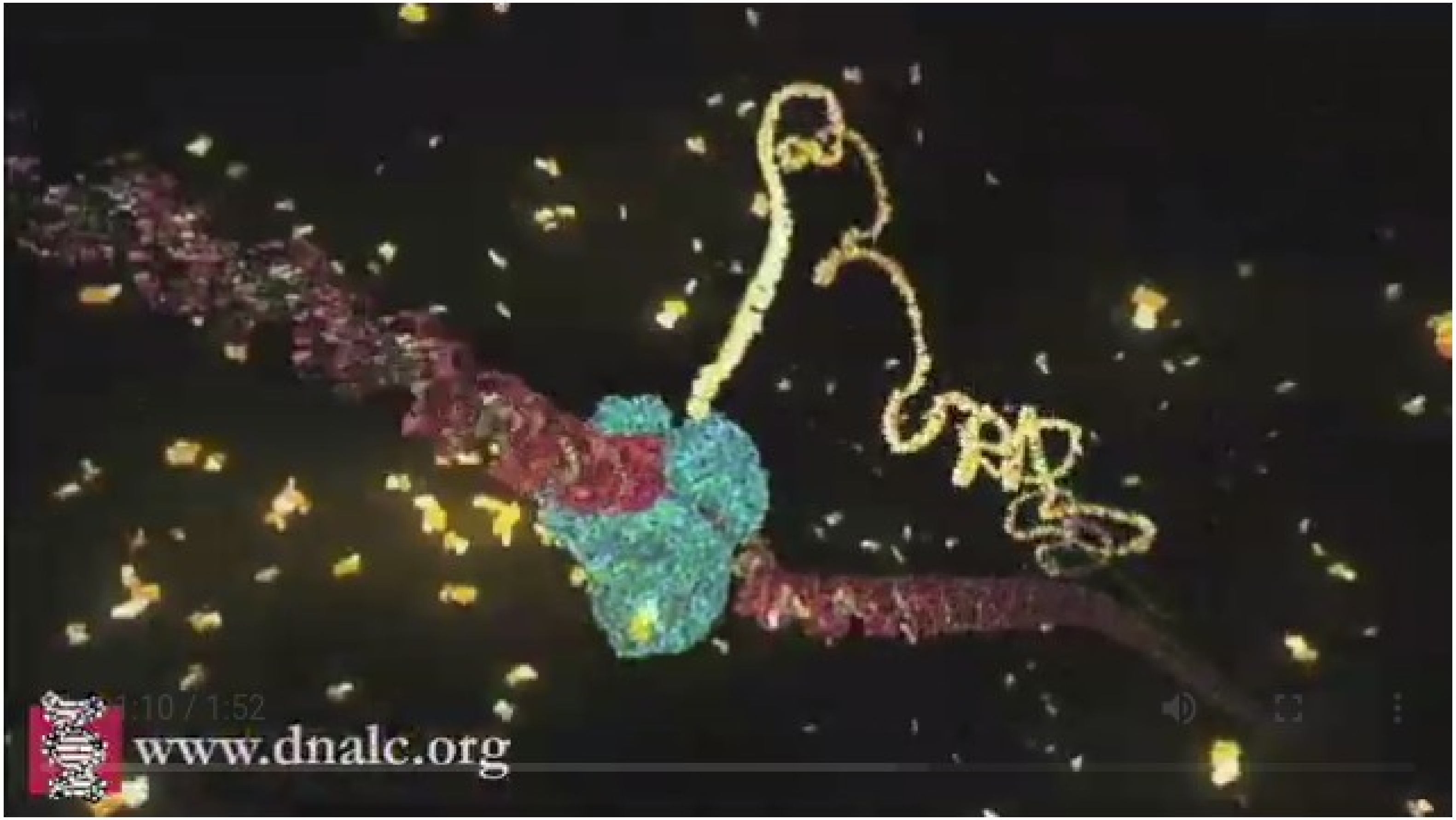
… <promoter> **Go Gators!** <terminator> …



Transcription is initiated at a promoter sequence and ends at a terminator sequence. The transcript is synthesized in a 5′-to-3′ direction.

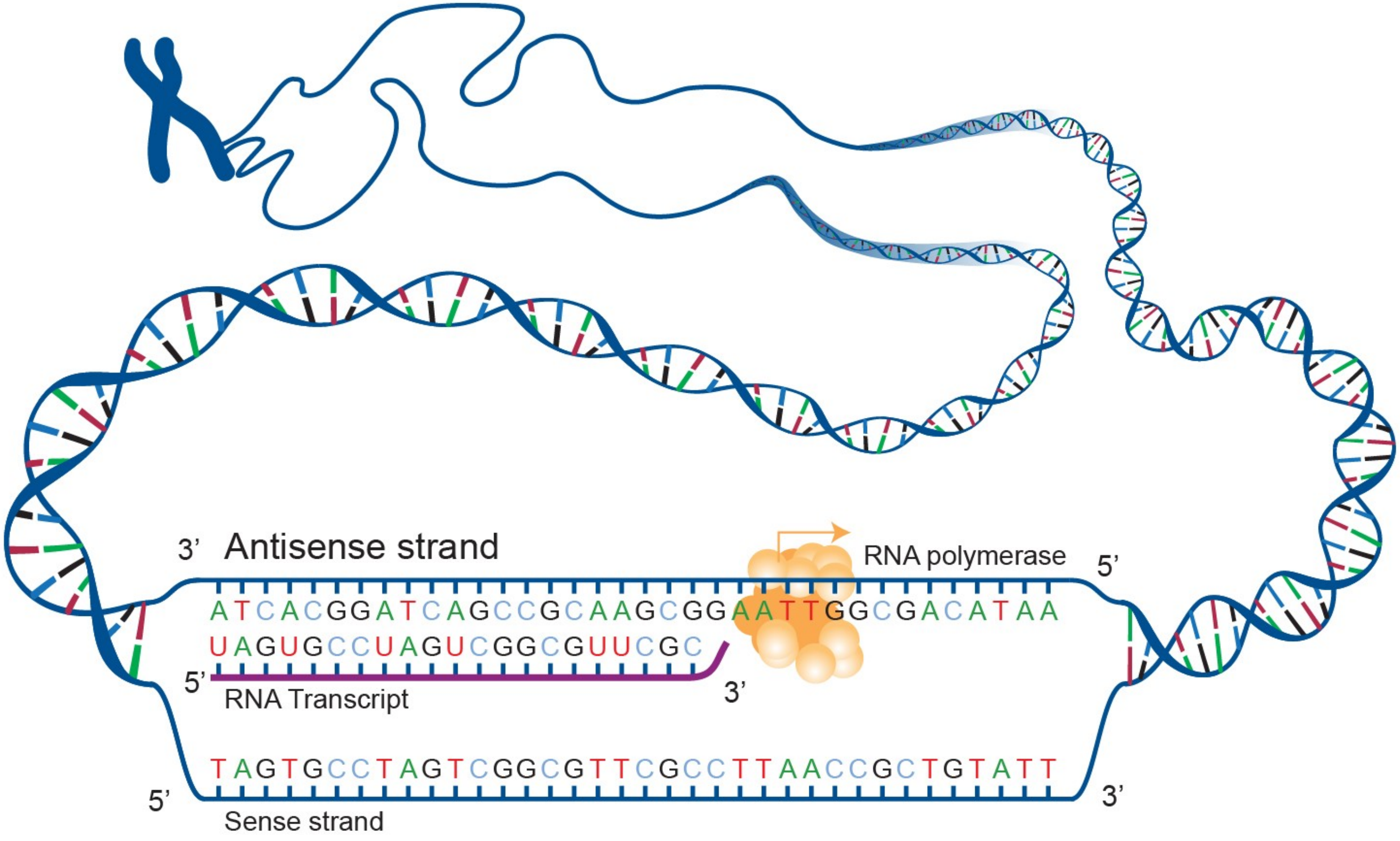
Both DNA strands serve as templates for transcription.

Transcription
RNA From DNA



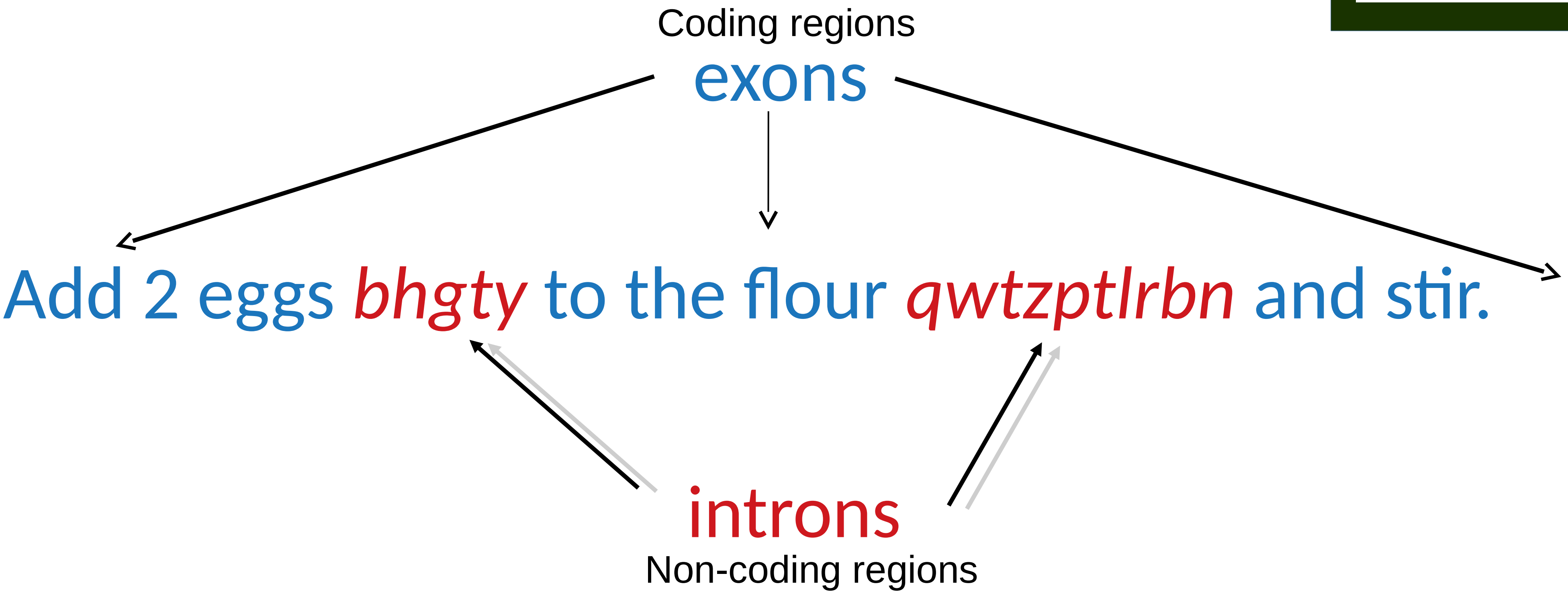
Cold Spring Harbor Laboratory link:
https://www.dnalc.org/resources/3d/12-transcription-basic.html

Sense and Antisense DNA



- Antisense is the non-coding DNA strand of a gene
- A cell uses antisense DNA strand as a template for producing messenger RNA (mRNA) that directs the synthesis of a protein.

Exon and Introns



Coding regions

exons

Add 2 eggs *bhgty* to the flour *qwtzptlrbn* and stir.

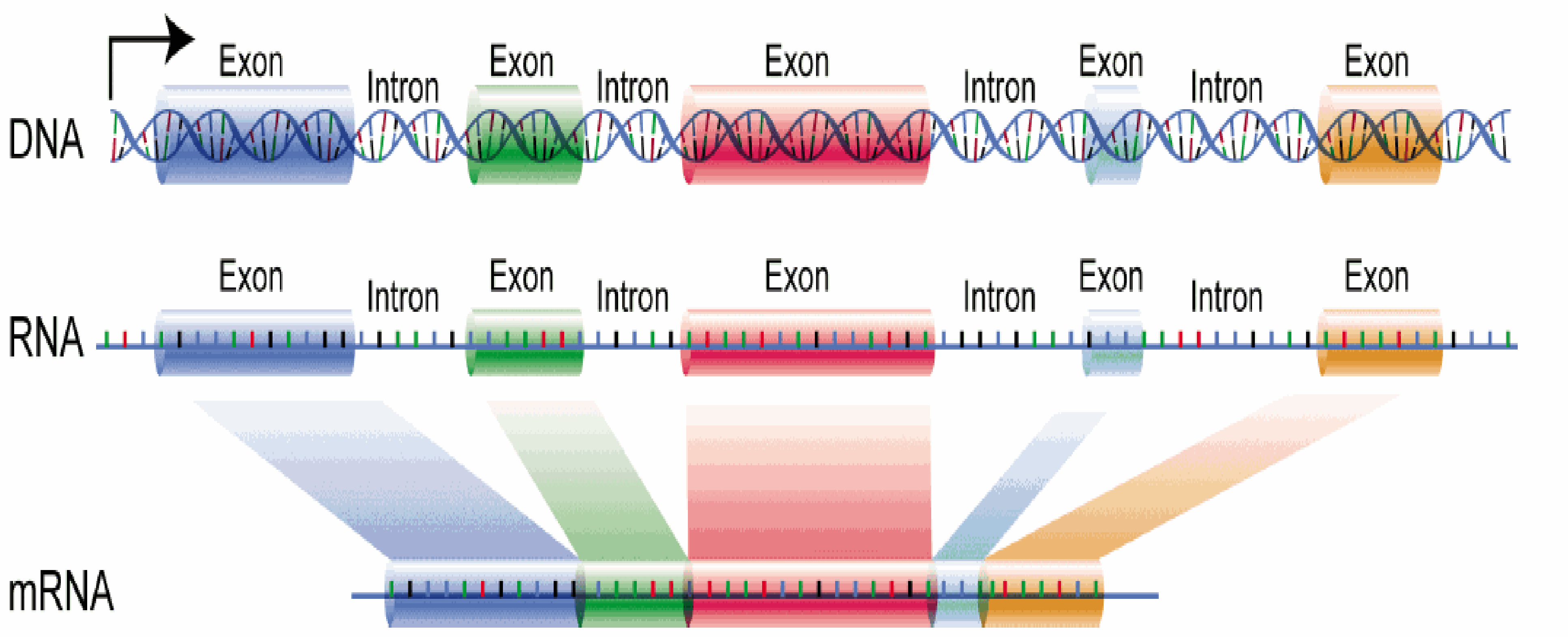
introns

Non-coding regions

Add 2 eggs to the flour and stir

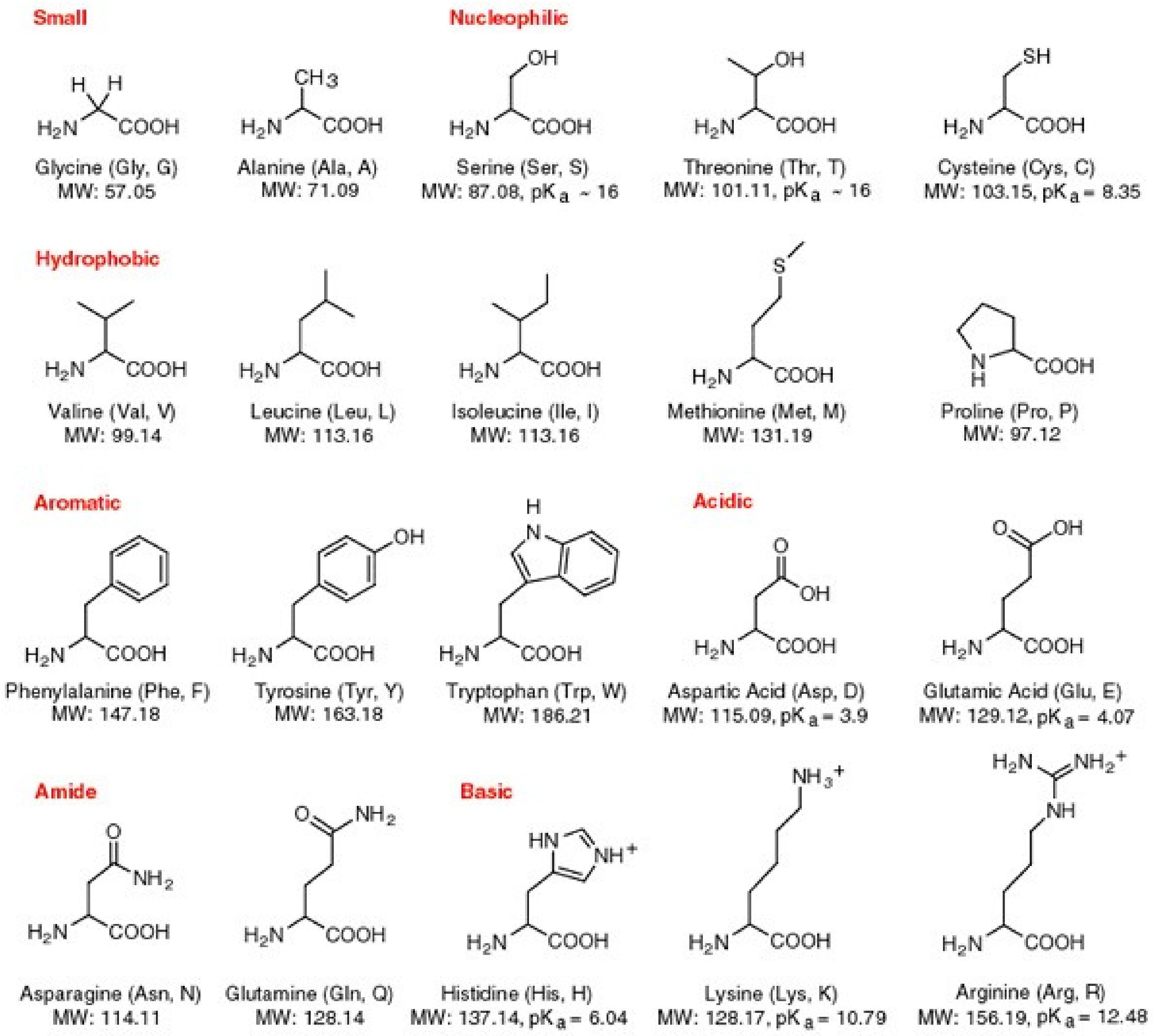
- In most eukaryotic genes, coding regions (exons) are interrupted by noncoding regions (introns). Introns do not contain the message and are removed from the RNA after transcription but prior to translation.
  During the process of RNA splicing, introns are removed and exons joined to form a contiguous coding sequence.

Splicing Exon Material



- Exons: a segment of a DNA or RNA molecule containing information coding for a protein or peptide sequence.
- Eukaryotic pre-mRNA contains exons and introns*
  - *some pre-mRNAs contain only one exon

Proteins are made of amino acids

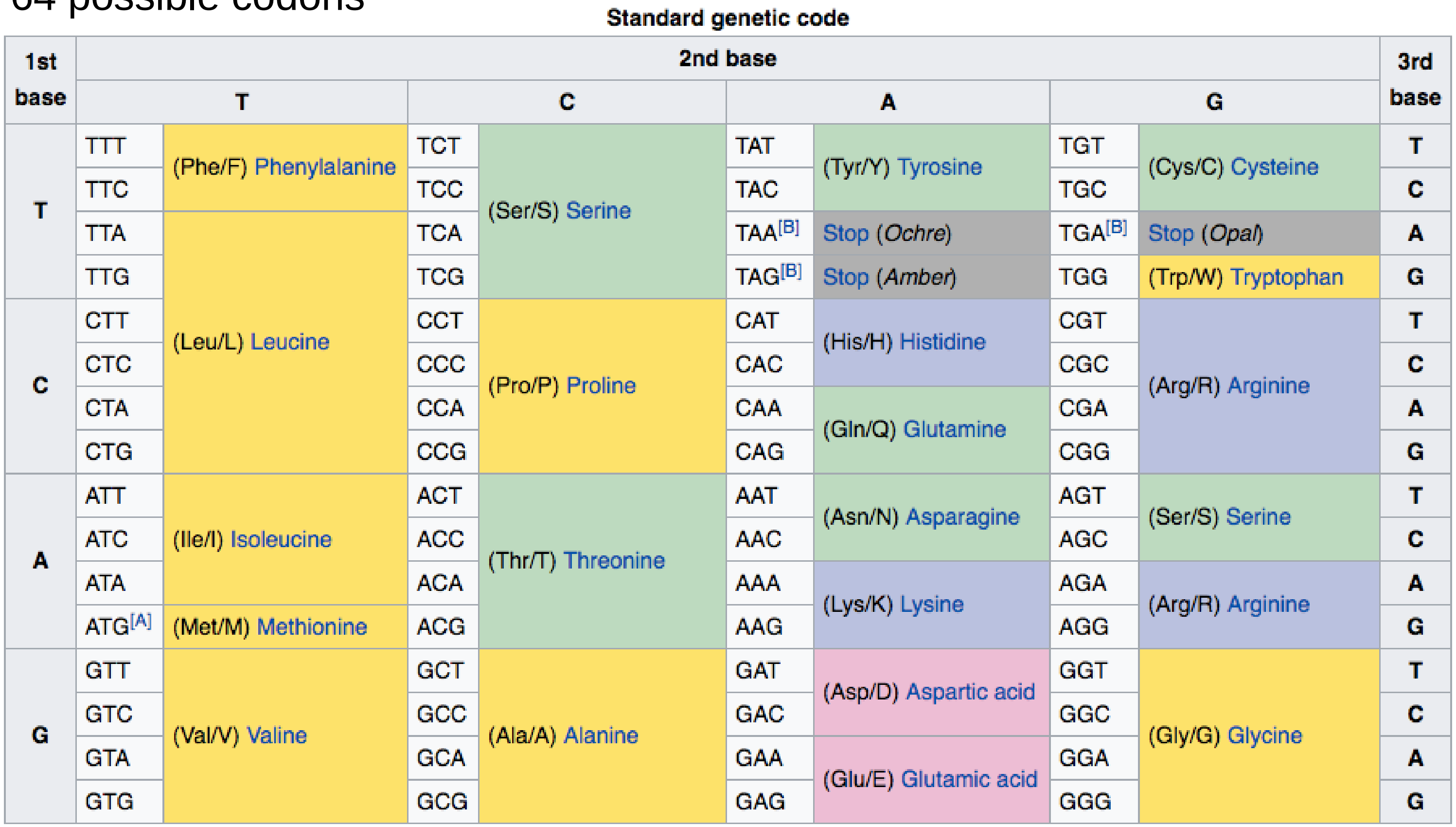


Translation: RNA into Protein

- Triplet code
  - Combinations of three nucleotides code for one amino acid
  - Three nucleotides = codon

- Redundancy
  - Sometimes >1 codon codes for same amino acid
  - 20 amino acids, 64 possible codons

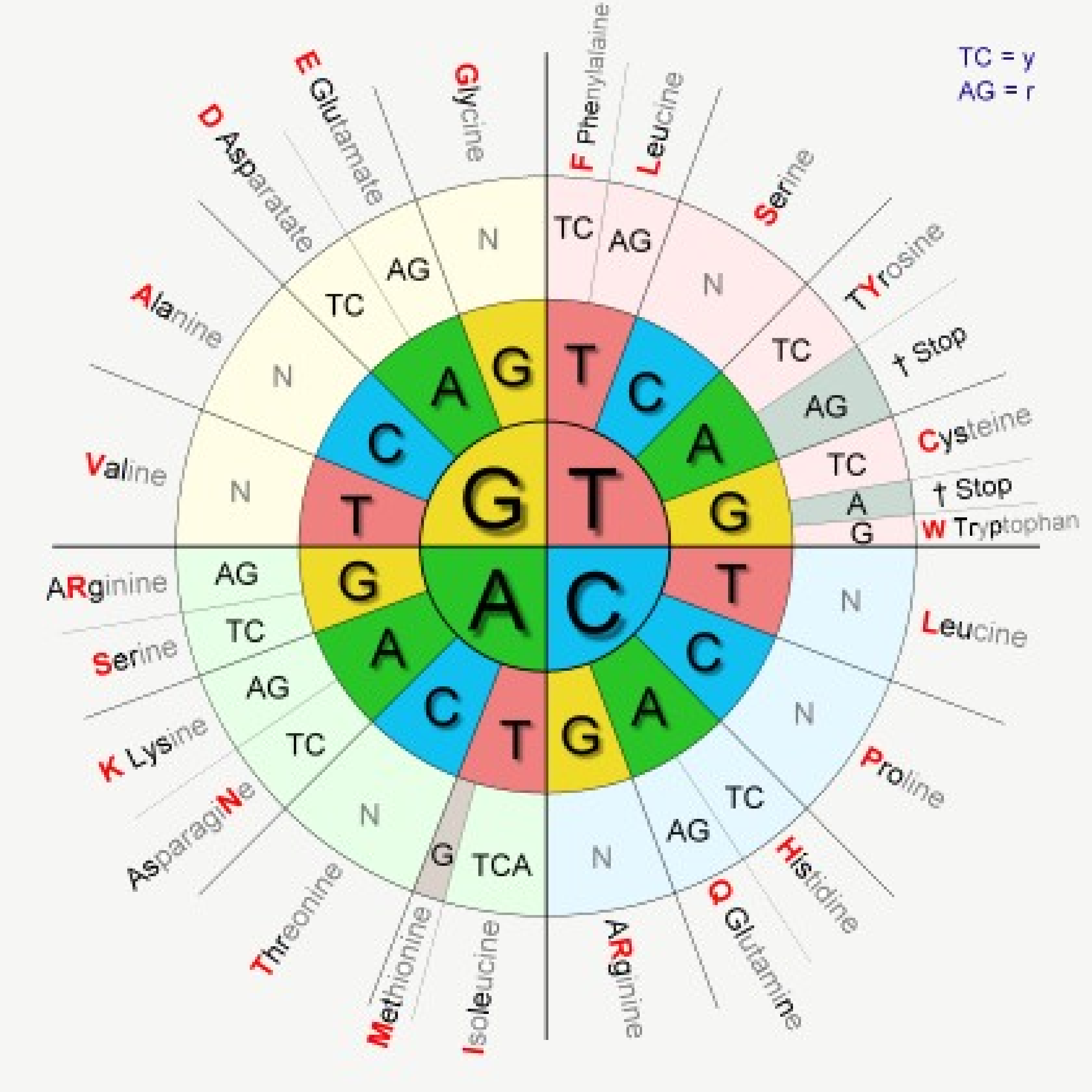
- Start and Stop codons
  - Often, ***Start*** codon is “AUG” (*methionine)*
  - Codons UAA, UAG, and UGA indicate the ***end*** of the transcript



Standard genetic code

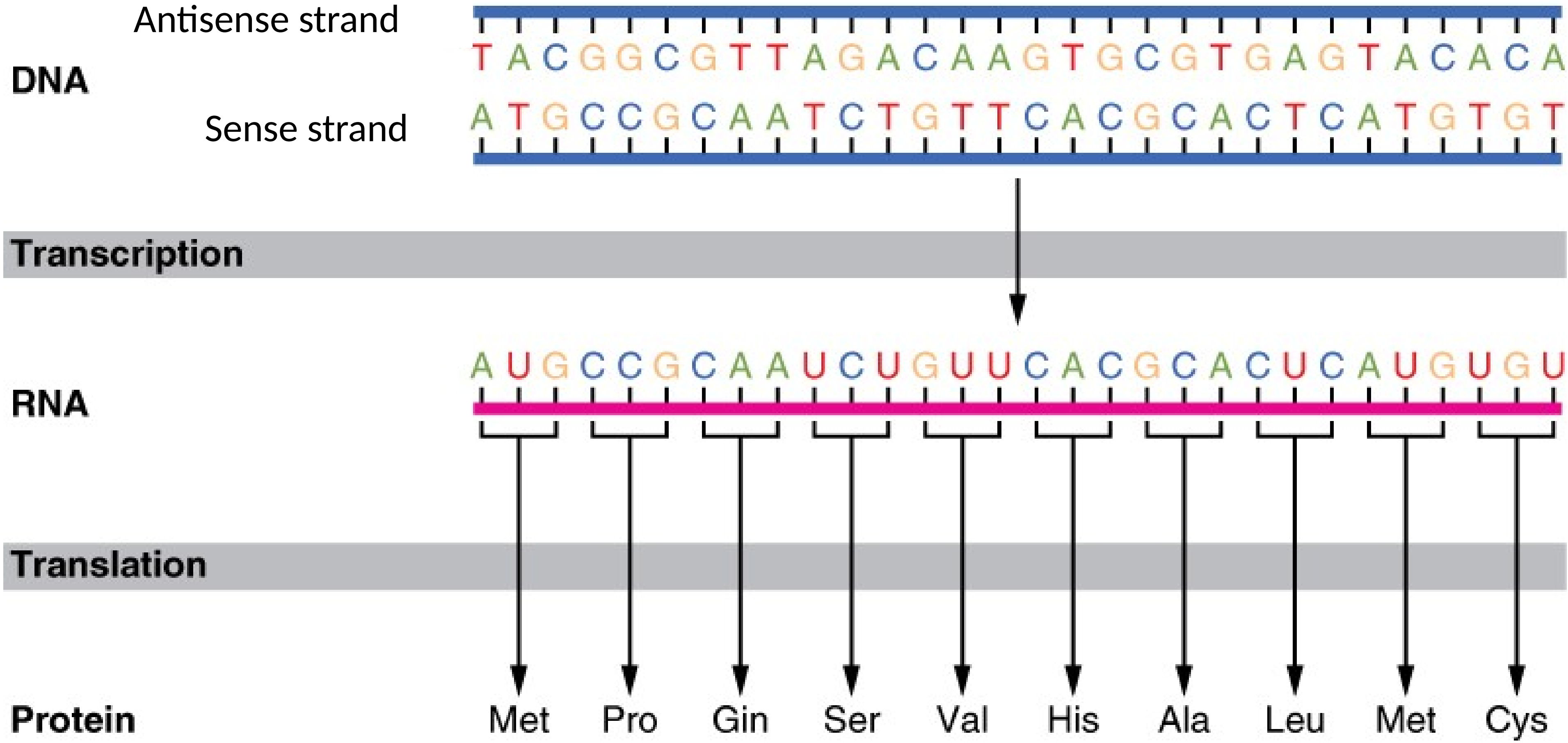
1st base	2nd base								3rd base
	T		C		A		G		
T	TTT	(Phe/F) Phenylalanine	TCT	(Ser/S) Serine	TAT	(Tyr/Y) Tyrosine	TGT	(Cys/C) Cysteine	T
	TTC		TCC		TAC		TGC		C
	TTA	(Leu/L) Leucine	TCA		TAA[B]	Stop (*Ochre*)	TGA[B]	Stop (*Opal*)	A
	TTG		TCG		TAG[B]	Stop (*Amber*)	TGG	(Trp/W) Tryptophan	G
C	CTT		CCT	(Pro/P) Proline	CAT	(His/H) Histidine	CGT	(Arg/R) Arginine	T
	CTC		CCC		CAC		CGC		C
	CTA		CCA		CAA	(Gln/Q) Glutamine	CGA		A
	CTG		CCG		CAG		CGG		G
A	ATT	(Ile/I) Isoleucine	ACT	(Thr/T) Threonine	AAT	(Asn/N) Asparagine	AGT	(Ser/S) Serine	T
	ATC		ACC		AAC		AGC		C
	ATA		ACA		AAA	(Lys/K) Lysine	AGA	(Arg/R) Arginine	A
	ATG[A]	(Met/M) Methionine	ACG		AAG		AGG		G
G	GTT	(Val/V) Valine	GCT	(Ala/A) Alanine	GAT	(Asp/D) Aspartic acid	GGT	(Gly/G) Glycine	T
	GTC		GCC		GAC		GGC		C
	GTA		GCA		GAA	(Glu/E) Glutamic acid	GGA		A
	GTG		GCG		GAG		GGG		G

Another Triplet Table



Translation

- The information from DNA is rewritten in a new language: RNA

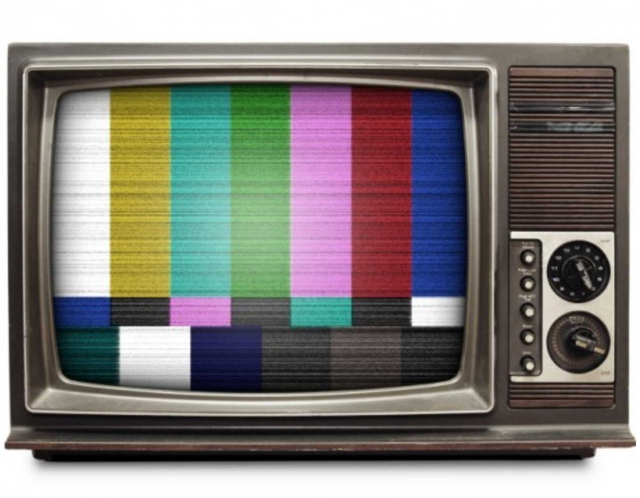

Antisense strand

Sense strand

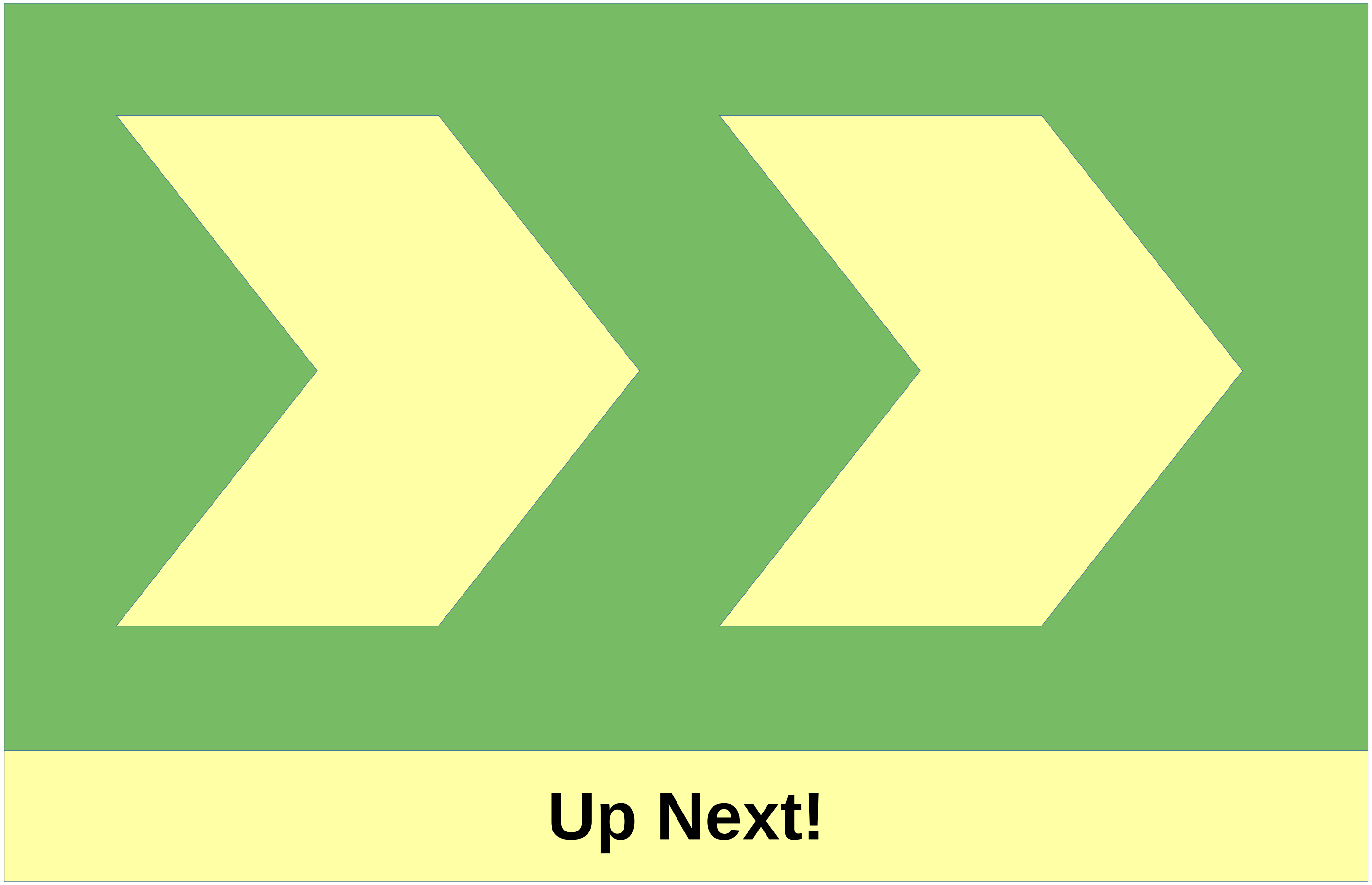
DNA
T A C G G C G T T A G A C A A G T G C G T G A G T A C A C A
A T G C C G C A A T C T G T T C A C G C A C T C A T G T G T
Transcription
RNA
A U G C C G C A A U C U G U U C A C G C A C U C A U G U G U
Translation
Protein
Met
Pro
Gin
Ser
Val
His
Ala
Leu
Met
Cys

Translation Videos

- mRNA Translation (2 mins)
  - https://www.youtube.com/watch?v=8dsTvBaUMvw
- Protein Synthesis and the Lean, Mean Ribosome Machines (7 mins)
  - https://www.youtube.com/watch?v=h5mJbP23Buo
- From DNA to protein - 3D, 2.5 mins)
  - https://www.youtube.com/watch?v=gG7uCskUOrA



Bring the Code!



Up Next!

Please Install
Your Software

- We will be using Git and GitHub. Please setup your account at https://github.com/ and also download a Git client software from https://git-scm.com/downloads
- We will also be using the Atom editor to write code. Please download and install your editor from https://atom.io/
- For most labs, we will be using Docker. Please download and install your Docker Desktop installation (note: not the Docker ToolBox) from https://www.docker.com/. Help: https://hub.docker.com/
- If necessary, please help each other to install this software. Or see the department’s Technical Leaders with questions.

File: Dockerfile

(for Docker Desktop)

Creates your container to run your Python code.

```
FROM ubuntu:20.04

RUN apt-get update && apt-get -y
install git htop vim python3 python3-pip

RUN \
pip3 install --upgrade pip


WORKDIR /root

CMD ["bash"]
```

Script files

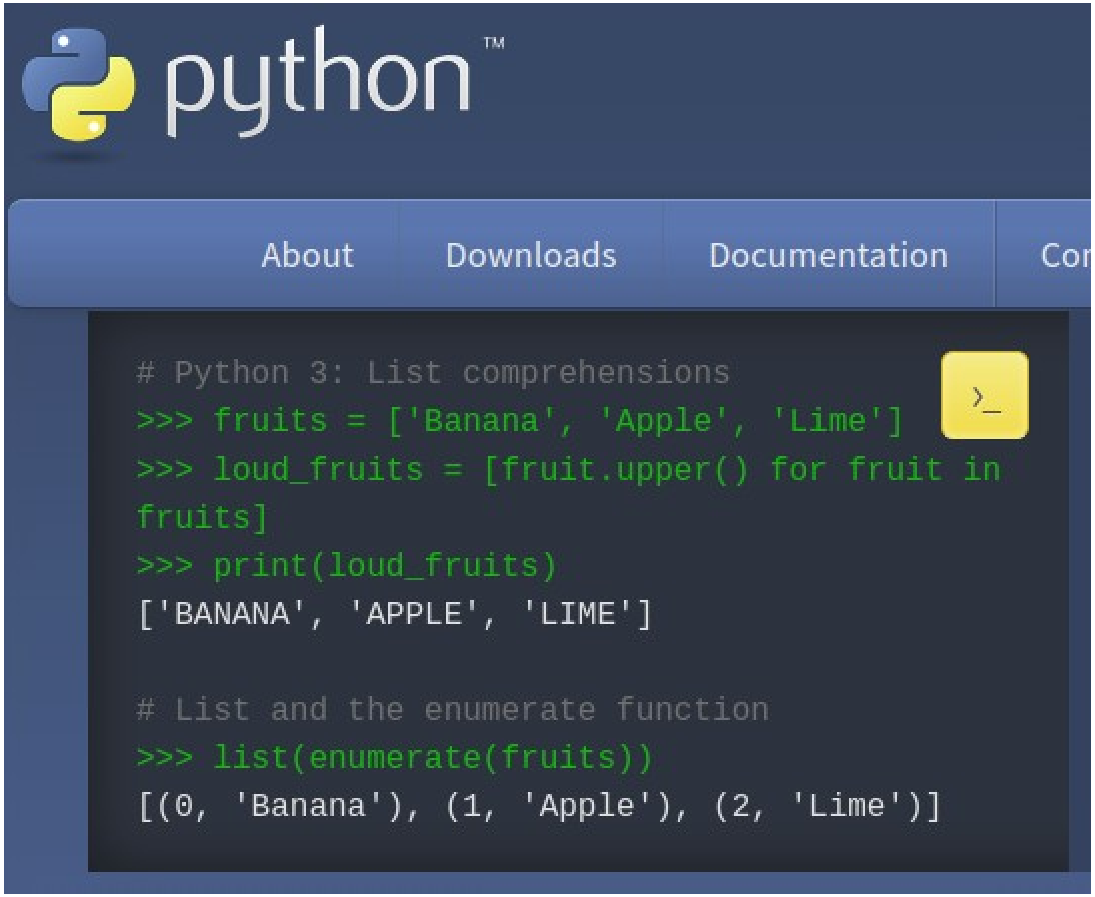
(for Docker Desktop)

Automatically create and run your containers!

Builder scripts:
- build_linux.sh
- build_macOS.sh
- build_win.bat

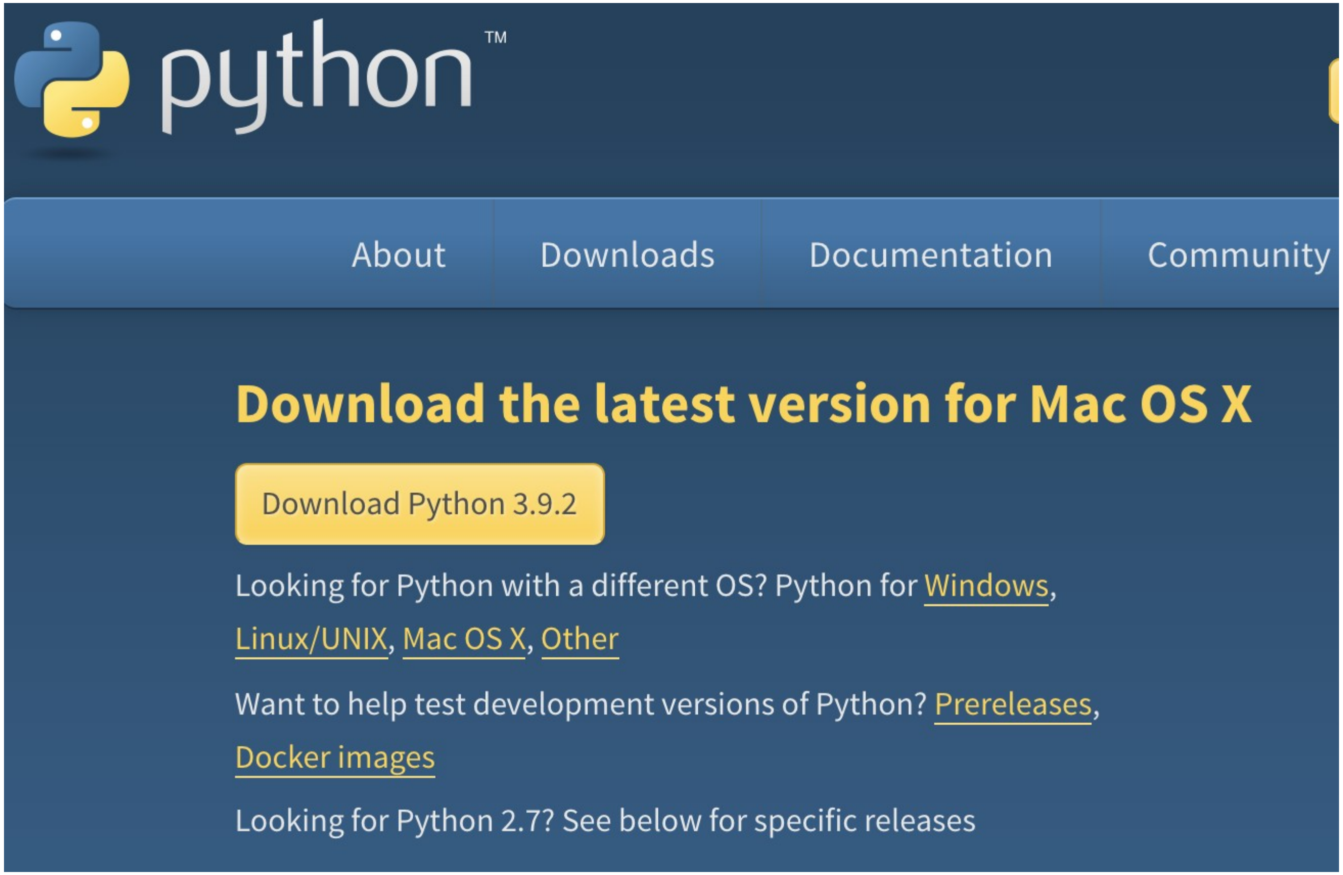
Executer scripts:
- run_linux.sh
- run_macOS.sh
- run_win.bat

- *Or, try Python3 programming using an interactive shell from repl.it*
- Link: https://repl.it/languages/python3



```
# Python 3: List comprehensions
>>> fruits = ['Banana', 'Apple', 'Lime']
>>> loud_fruits = [fruit.upper() for fruit in
fruits]
>>> print(loud_fruits)
['BANANA', 'APPLE', 'LIME']

# List and the enumerate function
>>> list(enumerate(fruits))
[(0, 'Banana'), (1, 'Apple'), (2, 'Lime')]
```

- Some trouble to make Python3 work with Docker ToolBox
- Install and use Python3 however you want!
- Get Python3 from the Python Software Foundation
- Login `http://www.python.org/downloads`



Download the latest version for Mac OS X

Looking for Python with a different OS? Python for Windows, Linux/UNIX, Mac OS X, Other

Want to help test development versions of Python? Prereleases, Docker images

Looking for Python 2.7? See below for specific releases

- Download and install the version of Python3 for your OS being sure to add the PATH to the environmental variables (check the path option!)
- Ask questions if you have trouble installing the program
- Check with the installation material to learn how to launch

Python3

```
#Calculating values
3 / 4
2 * 6
3.1415 – 2.718
x = 1
y = 2
print(x+y)
result = x + y
print(“The result is :”,result)
```

Python3

```
# Integers, counting numbers
   num_int = 1

# Floats, decimals
   num_float = 3.1415

# Strings
   s_str = ” Hello World”

# Combining variables in print statements
   x_int = 1
   print(" The integer variable is :", x_int)

   num_float = 3.14
   print(" The float variable is :", num_float)

   s_str =("Hello World'')
   print(" The integer is equal to", s_str)
```

Calculate

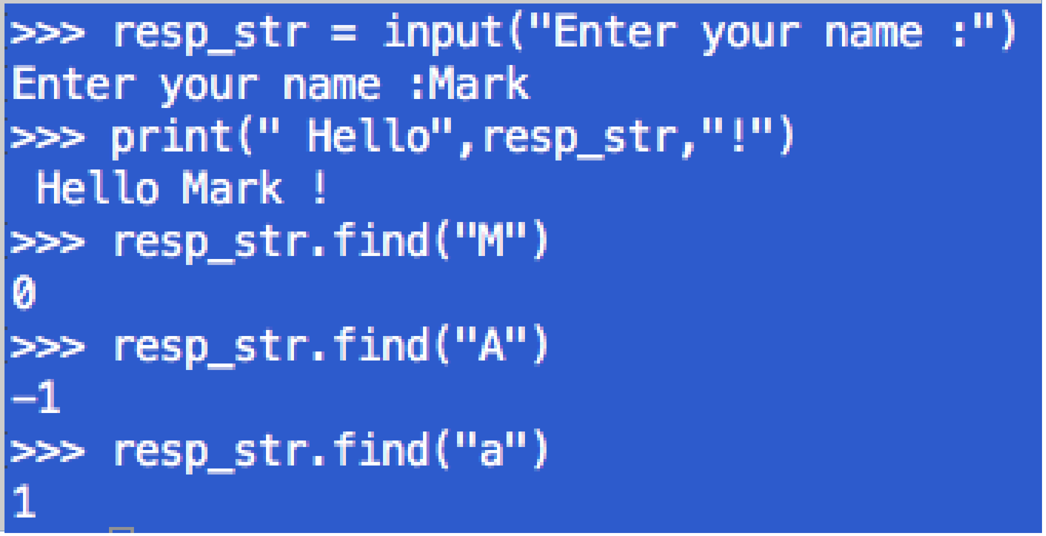
```
3 + 4 # Addition
3 − 4 # Subtraction
3 * 4 # Multiplication
3 / 4 # Division of 3 by 4

# Modulus; Returns the remainder from the
division 3 * *4
3%4

# Powers; raise three to the power of four
= 3*3*3*3
= 3^4
= pow(3,4)
```

Strings

```
# Remember each char of a string has own position
s_str = "ABC"
s_str[0] = 'A'
s_str[1] = 'B'
s_str[2] = 'C'
s_str[200] = ??

# Another way to iterate
# through a string using its length
for i_int in range(len(s_str)):
  print(s_str[i_int])
```

Counting and Finding

```
# Getting input from a user
resp_str = input(“Enter your name :”)
print(" Hello",resp_str,"!")

# Determine number of chars in a string
lengthOfName_int = len(resp_str)

# Find a subset-string in the string
resp_str.find("M")
resp_str.find("A")
resp_str.find("R")
resp_str.find("K")
resp_str.find("ARK")
```



```
>>> resp_str = input("Enter your name :")
Enter your name :Mark
>>> print(" Hello",resp_str,"!")
 Hello Mark !
>>> resp_str.find("M")
0
>>> resp_str.find("A")
-1
>>> resp_str.find("a")
1
```

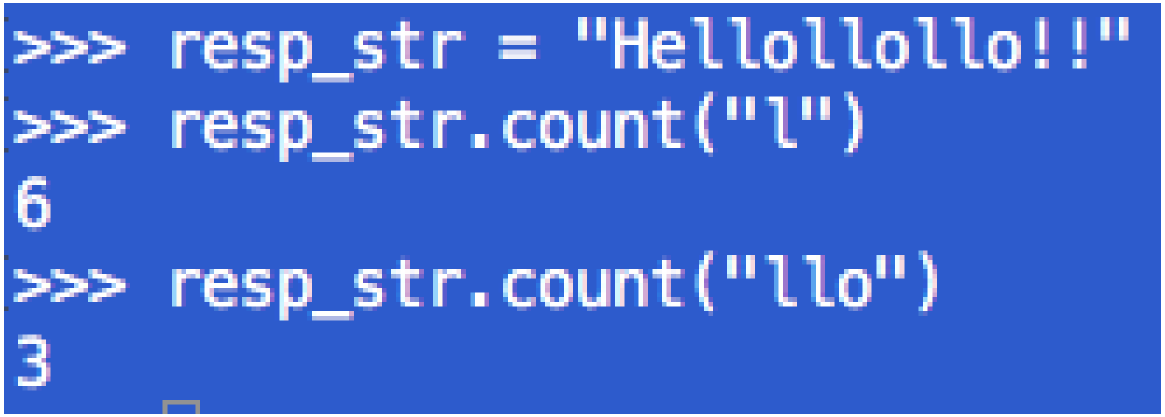
Counting and Finding

Find char occurrence in a string

```
resp_str = “Hello!!”
resp_str.count("H")
resp_str.count("l")
```

Find number of specific triples in string

```
resp_str = “Hellollollo!!”
resp_str.count("llo")
resp_str.find("llo")
```



```
>>> resp_str = "Hellollollo!!"
>>> resp_str.count("l")
6
>>> resp_str.count("llo")
3
```

A Short Program
Watch for tabs that define code blocks

```
print("Welcome to the program!")
prmpt_str = " Please enter your name :"

# place the string above into input statement
name_str = input(prmpt_str)
print(" Your name is :",name_str)
print(" And is <<",len(name_str),">> chars long!")

# print the chars on lines
print(" What are the characters in the string? ")

for i in range(len(name_str)):
    # note the tabs for this block!
    # we iterate through the positions in string
    print("  + char :",name_str[i])

# find out how many a's are in the name
numChar_int = name_str.count("a")
print(" The number of a's in your name :",numChar_int)
```

For this code, see file, pythonDemo.py, in your sandbox.

Consider This ...

- Work with your peers or by yourself
- Write a short program in Python3 accepts a short sequence of DNA (that you type in) and counts the occurrences of:
  - A’s
  - T’s
  - G’s
  - C’s
  - “AT”, “TA”
  - “GC” and “CG”

Now Consider This ...

- Now, go get a real piece of DNA and try out your code
- Link for organism: *Gordonia phage Orchid*, complete genome,
  - https://www.ncbi.nlm.nih.gov/nuccore/NC_030915.1?report=fasta
- What results did you find in terms of the pairs of AT’s, TA’s, GC’s and CG’s?
- Are the numbers of pairs similar or dissimilar?