



Amino

Acid Chart

Or if you prefer a table: https://en.wikipedia.org/wiki/DNA_and_RNA_codon_tables

Let's do a google search and you might find:

https://github.com/T101J/Translating_RNA_to_Protein

this is cool. A click on the code, and ... It kind of is what is in the homework ... that has promise. What is the license? OOps! no LICENSE file. Look at the code - they could have put the license in the file. Noope! That means that this is proprietary code - "All Rights Reserved".

Nice chart - but not in the class notes! This is *Proprietary* code and the nice blue chart is also!

The code. Useful to read - but it is not exactly what we want for class. Remember that we want to scan across to find a start-codon, then take 3 at a time and look them up. This assumes that you have already found the start - and just takes 3 at a time. It has the processing for the stop codon. That is good. So you can't copy this code but you can read it and learn from it.

There are about 5 other places where I found the exact same code on the web. Nice articles like: <https://towardsdatascience.com/starting-off-in-bioinformatics-rna-transcription-and-translation-aaa7a91db031> It has a good explanation of how to do this - but not exactly our requirements. (It also has the same chart - from what I can tell the chart is actually copyright a biomedical company.)

Note: going from DAN to RNA change 'T' to 'U'

First convert to one string

We have a list of strings. We need just one. "How do you convert a list of strings to a single string?"

Join them together. There is a string operation to do this called join. Join gives us the ability to join with a separator between each string.

```
#!/Users/philip/opt/anaconda3/bin/python

a = [ "aa", "bb", "cc" ]

print ( ", ".join(a) )
```

We can also join with a 0 length separator (this is what we will want in our code).

```
#!/Users/philip/opt/anaconda3/bin/python

a = [ "aa", "bb", "cc" ]

print ( "".join(a) )
```

So we start out with the data in the program:

```
#!/Users/philip/opt/anaconda3/bin/python

dna_input = [
    "taaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaacccta",
```



```
1: s = "aBBcBBd"
2: t = s.replace("B","-")
3: print ( t )
```

Use replace to remove a string.

```
1: # Remove Bs
2:
3: s = "aBBcBBd"
4: t = s.replace("B","")
5: print ( t )
```

replace with a longer string

```
1: # Replace Bs with a longer string.
2:
3: s = "aBBcBBd"
4: t = s.replace("B","--hi--")
5: print ( t )
```

Replace just the 1st occurrence

```
1: # Replace just the 1st B with a longer string.
2:
3: s = "aBBcBBd"
4: t = s.replace("B","--hi--",1)
5: print ( t )
```

Convert to upper/lower case

To convert to lower case

```
1: a = "This is A Mixed Case"
2: b = a.lower()
3: print ( b )
```

To convert to upper case

Functions for Lab 05

First is a function to take our input, convert from upper/lower case to lower, then replace the t's with u's.

Second is to use a dictionary lookup.

To create a dictionary in Python

```
d = { "name": 1, "name2", 2 }
```

or to lookup other strings

```
d = { "in1": "out1", "in2", "out2" }
```

So we can lookup our 3 letter codon with a dictionary.

```
#!/Users/philip/opt/anaconda3/bin/python
```

```
# RNA codon table
```

```
rna_codon = {
    "aaa" : "K", "aac" : "N", "aag" : "K", "aau" : "N", "aca" : "T", "acc" : "T", "acg"
    "acu" : "T", "aga" : "R", "agc" : "S", "agg" : "R", "agu" : "S", "aua" : "I", "auc"
    "aug" : "M", "auu" : "I", "caa" : "Q", "cac" : "H", "cag" : "Q", "cau" : "H", "cca"
    "ccc" : "P", "ccg" : "P", "ccu" : "P", "cga" : "R", "cgc" : "R", "cgg" : "R", "cgu"
    "cua" : "L", "cuc" : "L", "cug" : "L", "cuu" : "L", "gaa" : "E", "gac" : "D", "gag"
    "gau" : "D", "gca" : "A", "gcc" : "A", "gcg" : "A", "gcu" : "A", "gga" : "G", "ggc"
    "ggg" : "G", "ggu" : "G", "gua" : "V", "guc" : "V", "gug" : "V", "guu" : "V", "uaa"
    "uac" : "Y", "uag" : ".", "uau" : "Y", "uca" : "S", "ucc" : "S", "ucg" : "S", "ucu"
    "uga" : ".", "ugc" : "C", "ugg" : "W", "ugu" : "C", "uua" : "L", "uuc" : "F", "uug"
    "uuu" : "F"
}
```

We can see if a value is in the lookup table with an "if"

```

from rna_lookup import rna_codon

if "xyz" in rna_codon:
    print ( "found" )
else:
    print ( "not found" )

if "gac" in rna_codon:
    print ( "found value = {}".format(rna_codon["gac"] ) )
else:
    print ( "not found" )

```

A Program to go from DNA to Amino Acids.

Program State

```

1: #!/Users/philip/opt/anaconda3/bin/python
2:
3: import conv_t_to_u
4: import rna_lookup
5:
6: dna_input = [
7: "taaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaacccta"
8: "accctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaacc"
9: "cctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaacc"
10: "taaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaacccta"
11: "caaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaacccta"
12: "taaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaaccctaacccta"
13: "AGCCGGCCCCGCCCGGGTCTGACCTGAGGAGAACTGTGCTCCGCCTTCAGAGTACCACCGAAATCTGTGCAGAGGAc"
14: "aacgcagctccgccctcgcggtGCTCtccgggtctgtgctgaggagaacgCAACTCCGCCGTTGCAAAGGCGcgccgcgc"
15: "cggcgcgaggcgagagaggcgcgccgcgaggcgagagaggcgUGAcgcgccggcgagggcgagagaggcg"
16: "gccgcgcccggcgagggcgagagaggcgcgccgcgaggcgagagaggcgcgccgcgaggcgagggcgaga"
17: ]
18:
19: amino_string = ""
20:
21: rna = conv_t_to_u.conv_t_to_u ( "".join(dna_input) )
22:
23: st = "before"
24: i = 0
25: while i < len(rna)-2:
26:     three = rna[i:i+3]
27:     # print ( "st {} three >{}< at i={}".format(st, three,i))
28:     amino = rna_lookup.rna_to_amino_acid(three)
29:     if amino == '!':
30:         print ( "Invalid 3 letter sequence {} at {}".format(three, i) )
31:         break

```

```
31:         break
32:     if st == "before" and amino == "M" :
33:         st = "encode"
34:         i = i + 3
35:     elif st == "encode" and amino == ".":
36:         st = "before"
37:         print ("Protein : ", amino_string)
38:         i = i + 3
39:     elif st == "encode" :
40:         amino_string = amino_string + amino
41:         i = i + 3
42:     else:
43:         i = i + 1
44:
```

Copyright

Copyright © University of Wyoming, 2021.