**Write-Up:**

To verify that you tested your program, create a word or pdf file with the information below. Include your name(s) and lab number in the file. Include this file in your .zip file to moodle when you are finished.

**Testing and Evaluation (45 points, approx. 5 pts/problem):**

1. Test your calcGC function on the entire yeast genome. (yeastGenome.txt on Moodle).
   1. What is the GC content of yeast? [Be sure your program file and the txt file are in the same folder for the program to work.]
   2. What is the GC content of E. coli? (eColiGenome.txt on Moodle)
   3. What is the GC content of human chromosome 22? (human22.txt on Moodle – this is a big file)
2. Create your own FASTA formatted text file with bases other than ACGT and verify that your function for finding ambiguous characters works. Include a printout of your created FASTA file and what your function produces in your write-up here.
3. Execute the printCodons function on the rad55 gene in yeast. The sequence can be found here: <http://yeastmine.yeastgenome.org/yeastmine/begin.do>

Search for rad55. The first result should be a FASTA file. Include the first 10 lines of output from your program in your write-up here. If you do not like printing to the screen, write the codons to an output file instead. The first codon is ‘atg’, the second is ‘tcg’, etc. You will see soon why breaking a gene into codons will be useful for genomic analysis.

1. Test the createFASTA function with a string containing between 70 to 90 characters. Paste the content of the text file created in your write-up here.
2. Find TATAAA subsequences in yeast and eColi.

Choose one (or more) of the 16 yeast chromosomes. They are located in the yeast genome database below. Be sure to save this as a text file in the same folder as your python script. [http://downloads.yeastgenome.org/sequence/S288C\_reference/chromosomes/fasta/#](http://downloads.yeastgenome.org/sequence/S288C_reference/chromosomes/fasta/)

* 1. Include the chromosome # you chose in your write-up and the first 10 locations of TATA boxes on the forward strand in your write-up. (Since this function takes a sequence as a parameter, you’ll need to convert the file to a sequence using the function in exercise 0. Store this as a string seq and then execute your function on seq.)
  2. eColi (find the first 10 locations and include these in your write-up)

|  |  |
| --- | --- |
| **TATAAA box** | **First 10 locations (forward)** |
| Yeast chromosome #: \_\_\_\_\_\_\_\_ |  |
| Ecoli genome |  |

1. Find TATAAA subsequences on the reverse complement strand of the following organisms:
   1. Yeast chromosome you chose for part 5. Copy and paste the **last** 10 locations on reverse complement (first 10 on forward strand) in your write-up here.
   2. eColi (include the **last** 10 locations on reverse complement)

|  |  |
| --- | --- |
| **TATAAA box** | **Last 10 locations (reverse complement)** |
| Yeast chromosome #: \_\_\_\_\_\_\_\_ |  |
| Ecoli genome |  |

1. Find degenerate TATA box subsequences in the yeast chromosome you chose.
   1. For each of the 3 degenerate subsequences, create a list of the first 10 locations of the subsequence on the forward strand.

|  |  |
| --- | --- |
| **Degenerate TATA box**  **Yeast chromosome #:** | **First 10 locations (forward)** |
| TATACA |  |
| TATAGA |  |
| TATATA |  |

* 1. For each of the 3 degenerate subsequences, create a list of the first 10 locations on the reverse complement strand.

|  |  |
| --- | --- |
| **Degenerate TATA box**  **Yeast chromosome #:** | **First 10 locations (reverse)** |
| TATACA |  |
| TATAGA |  |
| TATATA |  |

1. Create your own DNA substring of length 5 – 8 bases. Does your yeast chromosome have this subsequence on the forward strand? If so, at what positions? Include the subsequence and the first 10 positions on the forward strand in your report.
2. What was the most challenging aspect of this lab? What did you and your partner learn from completing it?

**Appendix A: Statement of Authored Work**

The code and lab report are the work of the authors of this report. All external consultation resources and information are cited appropriately. Both authors contributed to this work.

**Appendix B: Code (40 points for correctness and proper style):**

Copy and paste the python code you wrote for this lab (use courier font size 9) in Appendix A. Your code should be properly commented and neat. Use fixed-width text font, so the characters/indentations align.