**CSC 314, Exam II Review**

**Note:** This review is not comprehensive, but contains several practice problems to help prepare for Exam II. In addition to these practice problems, you should make sure to understand all labs assigned since the first exam.

## *1. Sequence Database questions*

Look at the GenBank entry with accession number NM_001185098.1 to answer the questions below:

a. What is the length of this sequence?
b. When was the sequence last modified?
c. How many exons does this gene have?
d. What position marks the beginning and end of the poly-adenylation signal sequence, and what is this sequence?
e. What are the first nine nucleotides in the coding sequence (CDS), and the corresponding amino acids (you can give the 1-letter amino acid codes)?

## *2. Sequence Alignments*

For (a) and (b), use a linear gap penalty of 4 points, a match score of +5 points, and a mismatch score of -1 point.

a. Find the optimal global alignment and optimal global alignment score for the words *handy* and *say*. You must show your dynamic programming matrix to receive credit.

b. Find the optimal local alignment and optimal local alignment score between the words *stars* and *that*. You must show your dynamic programming matrix to receive credit.

c. Using the BLOSUM-62 matrix, a gap opening penalty of 5, and a gap extension penalty of 1, find the score of the *semiglobal* alignment given below (Recall that semiglobal alignments do not penalize gaps at the beginning or end of the alignment).

```
F R I D A - - Y
- - P - A R T Y
```

## 3. *Coding questions*

### *Coding question #1*

The regular expression corresponding to a potential coding sequence (CDS) is given by: "ATG(?:.{3})*?(?:TAG|TAA|TGA)"

Note: it is not necessary to understand the regular expression (this is much more advanced than the examples covered in class). For an explanation, put the regular expression into https://regex101.com/ and look at the description below:

- ATG – the start codon, ATG
- (?:.{3})*?- any number of codons (0 or more) (will match as few times as possible, i.e., *non-greedy*)
- (?:TAG|TAA|TGA)– matches any of the stop codons

Technical note #1: Normally parentheses denote a *capturing group*, and the match to this pattern is returned; to prevent this, we use (?:) which makes the group *non-capturing*.

Technical note #2: the asterisk (*) will match the preceding pattern as many times as possible, which is known as *greedy* evaluation. To match a pattern as *few* times as possible, follow the asterisk with a question mark (*?), which is known as *lazy* evaluation.

Technical note #3: the regular expression above will not find a potential CDS if it is within a larger one. For example, ATGATGTGA contains two potential CDS (ATGATGTGA and ATGTGA), but only the larger one will be returned. However, we could use a *positive lookahead* (?=) to handle these cases as well.

**Question:** Suppose that a file called *sequences.fasta* contains a large number of sequences in FASTA format. Write a python script that generates a list of the sequences that contain at least one possible CDS.

### *Coding question #2*

Suppose that the header of a FASTA sequence is of the form: *Human_GeneName* (e.g., *Human_TP53*). If the object *ids* contains a list of FASTA headers, use list comprehension to create a list that contains only the gene names.

4. ***BLAST***. The protein sequence for the C. elegans gene F55F8.2 can be found at accession NP_491652. BLAST this protein to find similar proteins in humans (limit your analysis to *Homo sapiens*). Note: the protein DDX24 is a known human ortholog of this worm gene, based on the fact that there is a high similarity.

    a. What is the % identity and % similarity of the alignment between F55F8.2 and DDX24?

    b. What pfam domains are present in this protein? Based on these domains, what do you infer about the function of F55F8.2?