

## 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, as well as the group project.

### 1. *Sequence Database questions*

(Start at GenBank: <https://www.ncbi.nlm.nih.gov/genbank/>)

- a. How many RefSeq molecules are there associated with the keyword insulin?
- b. How many of these molecules are from humans (*Homo sapiens*)?
- c. How many RefSeq entries are for the insulin gene INS in humans?
- d. The second hit from the above search is the gene with accession number NM\_001185098. How many exons does this gene have?
- e. What are the first nine nucleotides in the coding sequence (CDS), and the corresponding amino acids?

### 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. Using the UCSC table browser (<http://genome.ucsc.edu>), find the first 5 nucleotides of the promoter region of the *C. elegans* transcript Y51F10.10e. Assume that the promoter region begins 1000 base pairs upstream of the gene.
4. The regular expression corresponding to a standard coding sequence (CDS) is given by: "ATG(?:[ACGT]{3})\*(?:TAA|TGA|TAG)"

Note: it is not necessary to understand the regular expression (which is more advanced than the previous examples we have seen. But for completeness, the regular expression can be interpreted as follows:

- ATG – the start codon, ATG
- (?:[ACGT]{3})\* - either 0 or more of any codon (any three nucleotides)
- (?:TAA|TGA|TAG) – any of the stop codons

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.

5. Suppose that 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, write python statements that extract the corresponding gene names, which are stored in a list.