**Homework Assignment 2**

*Due date and time: Wednesday, January 20th, 2016*

*Submit both hard and electronic copies (PDF, python script) of your assignment*

*You may work with other members of your group on homework assignments but what you write-up and turn in must be solely your individual intellectual contribution.*

**TASK 1**: *N-W Dynamic Programming* (**3 points**)

Use the [Needleman-Wunsch applet](http://baba.sourceforge.net/) discussed in class to evaluate the optimal alignments for any three pairs of nucleotide sequences (you are at liberty to make these sequences up but do so in the form of a thoughtful experiment by choosing sequences that resemble each other strongly, moderately, and poorly; for convenience, I would not recommend making them longer than 10 residues). Set the gap penalty (for each sequence) of -2. Set the off-diagonal elements in the score table to -1 to bring it in line with our oft-used scoring system in class. Describe your ‘experiment’ and results and comment on the accuracy of the alignments (note that the scoring system is not particularly sophisticated and has limitations). Are you convinced the N-W algorithm works or do you think additional modifications are in order?

**TASK 2:** *N-W vs. S-W* (**4 points**)

Use the **Needle.&Wunsch** and **Smith&Waterm.** [Java applets](http://baba.sourceforge.net/) to compare the following two sequences: S1: PSYHIPI; S2: PAYHIEEEPSWHI. Use a gap score of -6 and the PAM250 matrix for matches and mismatches (you can select this from the pull-down in the Score Table). Comment on the alignments returned by the two algorithms. What can you infer about the evolutionary history of the two sequences? Are there other significant, non-overlapping, local alignment(s) that the applet could be programmed to return? [**NB:** *the applet is a learning tool and not a tool that you would use for aligning real-world sequences*]

**TASK 3:** *Substitution (aka Scoring) Matrices* (**4 points**)

In class, we saw how to construct a substitution matrix from scratch. I now want you to construct a simple nucleotide substitution matrix for an evolutionary distance of 1 PAM, which corresponds to an amount of evolution that changes 1 in 100 nucleotides on average. Assume that all four nucleotides are present at equal frequencies (nA = nG = nC = nT), share identical relative mutabilities (mA = mG = mC = mT; i.e. fA = fG = fC = fT), and that transitions (i.e. A ↔ G mutations or C ↔ T mutations) are eight times more likely than transversions (i.e. A or G ↔ C or T mutations). Be sure to round-off your scores to the nearest integer. You may use the example worksheet that we used in class for these calculations, as all the formulae for translating frequencies to substitution scores are already embedded in the worksheet.

**TASK 4:** *PAM vs. BLOSUM* (**2 points**)

Using the online [PAM matrix calculator](http://www.bioinformatics.nl/tools/pam.html), calculate the PAM160 matrix and compare it with the BLOSUM62 matrix. For example, mutations of which residues are tolerated in the former versus the latter. Comment on the origin of these differences.

**TASK 5:** *Exercises in Python* (**7 points**)

1. Read/browse through Chapter 2 of the *Python for Biologists* textbook.
   1. Write a python script that takes the following nucleotide sequence (which is a primer sequence you are using for a PCR reaction) below:

GGCTATCTATAGATAGCTTCGAGT

and computes the melting temperature (Tm in ˚C) using the formula below:

Tm = 81.5 + 0.41\*(%GC) – (500/N)

where N is the number of nucleotides and %GC is the percentage GC content. [**Hint:** break this task down to computing the GC content, then the number of nucleotides in the sequence, and finally the Tm] (**2 points**)

* 1. You are having trouble getting the PCR to work and you suspect the primer above might be self-dimerizing (i.e. binding to itself via a self-complementary region in sequence). Write a python script to compute the reverse complement of this sequence and think of a way to use one of the Excel worksheets we used in class to confirm or refute this possibility. Describe your approach and your results. (**2 points**)

1. Write a python script to splice out the intron (in lower case and colored in blue) from the following transcript:

CTTGACTCAGGCTGTCTCTGATTATCATGGAGTCAGTACCCTTGATAGAAAGGAAATTCCTCCAAGGAGGAAGTTCGAAAAGGGTACATCTACACTGTTCTAcatacaaaacttacaatcagtcctcatgcaggcccctgccatgctggttctgttatattaacaggaacactttCATGCCATCACCTATACATGGCCAAAAAGGACTCAGTTCTCCCCCACACCCCTTTTTCTATCTCTCTGATGTCTAATATCAGAGTACATTCCTGTGCTCCTCTAACACTCAAAACTA

In addition to reporting the sequence of the resulting mature mRNA, your script should also report the coordinates of the exons (starting and ending residues in the format 1…xxx or yyy…zzz) and that of the sole intron. [**HINT**: Use the find method and a 10-residue string to locate the coordinates for the beginning of the intron and the second exon] (**3 points**)