Background

The goal of the project is to implement a computer program, whose specification (including input and output formats) is described below. Programs, along with presentations, will be evaluated competitively in the final exam. Each team will present their implementation and describe what they see as advantages/disadvantages of their approach at the beginning of the final session. At the end of the final you will be asked to rank all other teams' programs and presentations, and this peer review will form a part of the grade.
Each program must be submitted as a tarball named **project.tar.gz** containing an executable named **runproject** (plus all dependencies: modules, libraries, etc) that, when run, will **look for the named input files in the current working directory, and produce the named output file**.
Specifically, if your tarball is in the current working directory, the following sequence of shell commands should output your results:

```
   tar -xvzf project.tar.gz
   ./runproject
   cat output.fasta
```

Team memberships

You can work in teams of up to 4 people, but only 50% of your team members can have been in the same team on a previous project.

Inputs and outputs

Inputs (use the specified filenames and formats):

- **protein.fasta** An ungapped alignment of equal-length protein sequences in FASTA format.
- **sites.fasta** A set of restriction enzyme sites that must not occur in the output (DNA sequence in FASTA format). Neither the site nor its reverse complement must appear in the eventual output.
- **effector.fasta** An effector RNA sequence (FASTA format). See below for description and relevance of this.
- **codonfreq.txt** A codon frequency table (64 rows, 2 columns: “codon”, a 3-character RNA codon, & “frequency”, a non-negative real number)
- **params.txt** A one-line file containing a single numeric parameter
    - Line 1: *N* (the number of variant DNA sequences your program should generate)

Outputs (use the specified filename and format):

- **output.fasta** A file containing *N* DNA sequences (FASTA format)

Each output DNA sequence must satisfy the following criteria:

1. It must encode a bacterial protein-coding gene, with a basic bacterial promoter (including a Pribnow box), a Shine-Dalgarno sequence, a start codon, a stop codon, and an intrinsic terminator.
2. The DNA sequence must be in lower case, except for the start and stop codons which must be in upper case.
3. There must be a riboswitch such that (at room temperature) the Shine-Dalgarno sequence will only be exposed if the effector sequence is present.
4. The most conserved/variable sites in the alignment represented by **protein.fasta** should correspond to the most conserved/variable sites in the output.
5. The *N* encoded proteins should be as diverse as possible. (So, the percentage identity between any two of your proteins should be as low as possible.)
6. The relative codon usage for each amino acid should match the frequency table as closely as possible, compared to synonymous codons. (For example, given that your protein sequence has 10 glycine residues, the corresponding protein-coding DNA sequence must use the codons GGA, GGC, GGG and GGT to code for those glycines. The relative frequencies of each of these codons should be proportional to the frequencies of those same codons in **codonfreq.txt**.)
7. Identical codons, or synonymous codons differing only at the third position, should be spaced apart as much as possible.

Riboswitch Design

- Teams can attempt to design a DNA sequence with a functional riboswitch that uses the supplied effector for full credit (20pts).
- Teams can elect to take a 5pt deduction and define their own effector rather than using the one that will be supplied.
    - If students choose this route, they should supply the effector sequence as a separate file **owneffector.fasta** to be output along with **output.fasta** in the same directory.
- Note: For the purposes of this project we assume that for an effector to successfully bind, its exact complement or reverse complement has to be present in the sequence. Furthermore, if multiple complement or reverse complement sequences occur throughout the entire length of the sequence (even within the coding region), we assume the effector binds to them and those nucleotides are completely unavailable for base-pairing.
- Note: It's sufficient for a single nucleotide of the Shine-Dalgarno sequence to be base-paired for it to be considered inaccessible. All six nucleotides of the Shine-Dalgarno sequence must be non-base-paired for it to be considered accessible (a non-base-paired SD sequence that is exposed in the loop region of a stem-loop is considered accessible).

Scoring scheme

- Points will be awarded for fulfilling the project criteria, with an exact distribution to be announced during the project period.
- Points will be deducted for increased computation time. Programs running for longer than about a minute may be killed (depending on time constraints), so make sure your program outputs sequences (and flushes the output buffer) as soon as it discovers them.
- You may work in teams or individually. **The maximum team size is four people**. At most 50% of your team members can have been in the same team on a previous project. Individual contributions to the team effort must be clearly delineated.
- **Scoring distribution**: The following features will be checked for in your generated DNA sequences (max available points = 75):

1. Check output.fasta is valid FASTA format, if not valid, no further points awarded.
2. Check for Promoter region (5pts).
3. Check for Shine-Dalgarno sequence (5pts).
4. Check for Start Codon (5pts).
5. Check for Stop Codon (5pts).
6. Check for Terminator (5pts).
7. Check for functional Riboswitch (20pts: half if Shine-Dalgarno sequence exposed when effector present, half if exposed when effector absent).
8. Check for Restriction sites and their reverse complement (10pts).
9. Perform competitive ranking against other teams (up to 20pts depending on rank). The score used for this ranking will be a weighted sum of the following terms:

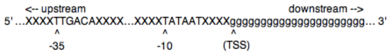
    - S / (sequence length), the length-normalized total column entropy in an alignment of all your encoded protein sequences (rewarding diversity in your generated protein sequences).
    - -M, a penalty term based on the maximum percentage identity between any one of your proteins and any protein from **protein.fasta** (rewarding generation of new sequences, rather than exact copies of the sequence in that file).
    - L / (sequence length), the length-normalized total log-likelihood of all your protein sequences using a position-specific weight matrix model based on the alignment in **protein.fasta** (rewarding identification of the conserved/variable sites).
    - ΔCodonFreq, the difference between the a codon's frequency in your output sequence and the target frequency, averaged across all codons (rewarding correct codon usage).
    - (Avg Space between similar codons) / (sequence length), the average space between identical/synonymous codons, averaged for all codons and divided by sequence length (rewarding avoidance of repeat codons).

- Note: For steps 1 through 8, the points will be split up across the desired number of sequences. Therefore, if the desired number of sequences is 4 (*N*=4), and only 3 of your 4 sequences have a functional riboswitch, you will be awarded 15 points.

Frequently Asked Questions

These are common questions about the design and function of your programs that you may find to be helpful. Check back occasionally for updates.

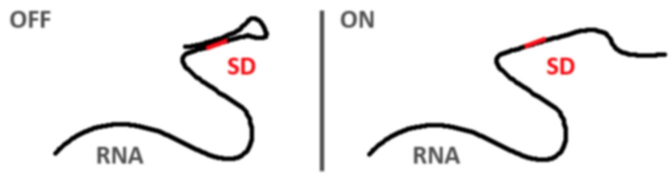
1. Should my promoter region contain specific sequences at the -10 and -35 sites?
Although there is natural variability in the sequence composition of prokaryote promoters, you should ensure that your DNA sequence contains a Pribnow box (specifically the sequence TATAAT) starting at the -10 position. The composition of the -35 sequence can have more flexibility in terms of composition and position. See example below:



```
     <-- upstream                                        downstream -->
5' ...XXXXTTGACAXXXXX...XXXXTATAATXXXXgggggggggggggggggggg... 3'
             ^                      ^               ^
            -35                    -10            (TSS)
```

For design purposes, the gene to be transcribed begins at the Transcription Start Site (TSS), exactly 10 nucleotides in the 3' direction from the start of the Pribnow box. In the figure, the gene is shown as "ggggg….."; note that these "g"'s are placeholder characters ("g" for "gene"). The actual sequence will not be all g's! You have to figure out what should go there, based on the criteria in this project description and any results from the molecular biology literature that you can bring to bear on the problem.

2. Does my sequence have to produce the hammerhead secondary structure as seen in lab/HW when there is no effector present and the Shine-Dalgarno sequence is base-paired (ie. OFF state)?
No, a hairpin structure would be adequate. See illustration below:



3. What are acceptable start codons?
AUG, GUG and UUG

4. Should the codon frequency determine the overall codon usage or simply the distribution across degenerate codons of individual amino acids?
The codon frequency table will give overall frequency of each of the 64 codons. You should use this information to get a similar relative frequency of codons that code for the same amino acid. Your overall codon frequency will be more directly impacted by the protein sequence input and the percent identity you want to achieve.

5. Do codon frequency files contain the overall frequency of all 64 codons or just the frequency of all degenerate codons that code for a specific amino acid? The codon frequency files will contain the overall frequency of all 64 codons with each line containing the nucleotide triplet, followed by a single space, followed by the frequency of that triplet *per thousand* codons (the sum of all the frequencies should be ~1000). See bellow:

```
UUU 22.2
UUC 15.9
UUA 13.8
UUG 13.0
... (56 lines omitted)
GGU 24.2
GGC 28.1
GGA  8.9
GGG 11.8
```

6. Can I use 3rd party software packages?
You may use the VIENNA package for RNA folding, if you wish (it's not mandatory). The test environment will have this package installed in the default $PATH.

7. There is variability in the prokaryotic terminator sequence, what are the requirements for terminator composition for this project?
You should ensure that your intrinsic terminator is composed of a CG-rich hairpin loop (at-least 4 base-pairs) followed by a poly-U tail (composed of at-least 8 sequential Uracil nucleotides). For the purposes of this project, transcription will be terminated immediately after the end of the 8+ Uracil nucleotides.