Homework 1: Midterm preparation

Due: Thursday, April 26, in class.

Problem 1. Reverse Complement.

What are the reverse complements of the following DNA sequences?

1. `ATGCAT`

2. `TGTGTATCAATCTG`

3. `ACTGATCTAAATGCTAA`

4. `AATTGGCCAATTGGCC`

Problem 2. Genetic Code.

Using the genetic code table (note: the table will be provided to you during the midterm exam as well), translate the following nucleotide sequences into the alphabet of amino acids, reporting the translations in each frame (use `"#"` for the stop codon).

1. `ATGTCCAACCATCAAGTA`

2. `AAATGAGCCCATGTGATAAA`

3. `TTGTGATGTTGTACCTATAGTA`

Problem 3. GC-percent.

We have a 50-character long DNA fragment, part of which is known to belong to a coding region. The coding region starts with the (only) occurrence of the `ATG` codon in the fragment and continues through the end of the fragment.

Compute the overall GC-percent of the entire fragment, as well as the GC-percent of the coding and non-coding regions. Do you think the latter two GC-percent numbers are significantly different (for your convenience, we mark positions 1, 11, 21, 31 and 41 in the string below)?

```
TCACTTATCAATGTTCATTCCAGTTGGTCATGATGGGAGGTACATCCGC

1         1         1         1         1
```

Problem 4. Codon Usage Bias.

Find the histogram of occurrences of Arginine (*Arg*, R) and the histogram of occurrences of Phenylalanine (*Phe*, F), in the following DNA fragment in the first frame (i.e., the first codon starts at the first nucleotide). Ignore all other amino acids (spaces are for your convenience).

```
AGT TTT CGT CGA CGT TTA TTC CGG CGA TTT GGT CGA TTC CGT TTT CGT CCC TTT CGT TAG
```

Problem 5. Codon Usage Bias.

Compute the optimal codon frequency for the DNA fragment from **Problem 4** given the following optimal codons for for Phenylalanine and Arginine respectively (ignore occurrences of codons for other amino acids).

1. `TTT` and `CGT`.
2. `TTC` and `CGA`.
3. `TTT` and `CGG`.

Problem 6. Codon Usage Bias

For the DNA sequence in Problem 4 and for each codon for Phenylalanine and Arginine compute its **RCSU**.

Problem 7. Codon Usage Bias

Compute the quantities S_{Arg} and S_{Phe} for Arginine and Phenylalanine for the DNA sequence in **Problem 4** (from the prep work for computing the Effective number of codons).

Problem 8. Enthropy

Compare the Shannon's information enthropy for the coding and non-coding regions for the DNA fragment from **Problem 3**.

Compare the Shannon's information enthropy for the two DNA fragments below:

```
S1:  ATGAATTAGGTTTTAAATTTAAATAG

S2:  ATGCTGTAGTCAGTTACCTGATCTAG
```

Problem 9. String Matching (KMP)

Compute the prefix function $\pi()$ (represent it as an array of values) for the following strings:

```
S1:   AGTTGTC

S2:   CATCATCAT

S3:   ACTACGTTTACTACTACAC

S4:   TTTTGTATTGTTTTGA

S5:   ATAGATACATAGATACATAG
```

Problem 10. String Matching (KMP).

Show how KMP algorithm will look for the occurrences of the string `P= ATGATCAT` in the string `S= ATCATCATATGATCATGATCATTTG`

Note: Here is a sample of how to do it. Let `P = CTCGC` and `S = ACTGACTCGCTA`. We show the work as follows.

The prefix function $\pi_P[i]$ is:

Position	1	2	3	4	5
Character	C	T	C	G	C
$\pi_P[i]$	0	0	1	0	1

The sequence of comparisons will look as follows (`x` shows mismatch, | shows match):

```
ACTGACTCGCTA
x
CTCGC

ACTGACTCGCTA
 ||x
 CTCGC

ACTGACTCGCTA
     ||||| ---> matched @ position 6
     CTCGC

ACTGACTCGCTA
        --> end of S reached.
        CTCGC
```

Problem 11. String Matching (Boyer-Moore).

Compute the bad character rule and good suffix rule shifts (arrays $R[1..|\Sigma|]$ and $GS[1..m]$) for the Boyer-Moore algorithm for the following strings $P[1..m]$:

P1: CATCAT

P2: GAAATGAT

P3: AGTCTGACTG

P4: TTTAAACCCTACTA

P5: GCTGTACTGACTGACTG

Problem 12. String Matching (Boyer-Moore).

Show the work of the Boyer-Moore Algorithm when finding all occurrences of the string AGTCTGACTG (string P3 from **Problem 11**) in the string

GATCATAGTACTGAGTCTGAGTGAGTCGTAGTATGGG

Problem 14. Overlap detection (Boyer-Moore).

Using the good suffix rule, compute the maximal overlap between the strings $S1$ and $S2$ below: Show all steps.

S1: ATGTAGGTAGGTGACCTCCCATGGTAGTACTAC
S2: AGTACTACAAAGTACAGAC

Problem 15. Suffix Trees.

Construct suffix trees for the following strings:

S1: ACTCTA

S2: GCCGCT

S3: CATCAGCATC

S4: ACACACAC

S5: GACTACGACT

Problem 16. Repeat search.

Modify the suffix tree for the string GACTACGACT to prepare it for the search for maximal repeated sequences. Find all maximal repeated sequences of length greater than 3.