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. TGTATCAATCTG

3. ACTGATCATAATGCTAA

4. AATTGCAATTGCCC

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. ATGTCACCACATCAAGTA

2. AAATGAGCCCATGTGATAAA

3. TTGTATGTGTACCTATAGTA
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)?

TCACCTATCAATGTTACATCCAGTTGCTATGATGGGAGGTACATCCGC

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 CCCTTT 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 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. Entropy

Compare the Shannon’s information entropy for the coding and non-coding regions for the DNA fragment from Problem 3.
Compare the Shannon’s information entropy for the two DNA fragments below:

S1: ATGAATTAGTTTTAAATTTAAATAG
S2: ATGCTGATCGTTACCTGATCTAG

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: ACTACGTTTACTACAC
S4: TTTTGTTTGTGTTTG
S5: ATAGATACATAGATACATAG

Problem 10. String Matching (KMP).

Show how KMP algorithm will look for the occurrences of the string \( P = \text{ATGATCAT} \) in the string \( S = \text{ATCATCATGATCATGATCATTTG} \).

**Note:** Here is a sample of how to do it. Let \( P = \text{CTGC} \) and \( S = \text{ACTGACTGCTA} \). We show the work as follows.

The prefix function \( \pi_P[i] \) is:

<table>
<thead>
<tr>
<th>Position</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Character</td>
<td>C</td>
<td>T</td>
<td>C</td>
<td>G</td>
<td>C</td>
</tr>
<tr>
<td>( \pi_P[i] )</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

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

ACTGACTGCTA
x
CTGC

ACTGACTGCTA
||x
CTGC

ACTGACTGCTA
|||x
CTGC

ACTGACTGCTA
||||x ---&gt; matched @ position 6
CTGC

ACTGACTGCTA
---&gt; end of S reached.
CTGC
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


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 $S_1$ and $S_2$ below: Show all steps.

- **S1**: ATGTAGGTAGGTGACCTCCATGGTAGTACTAC
- **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.