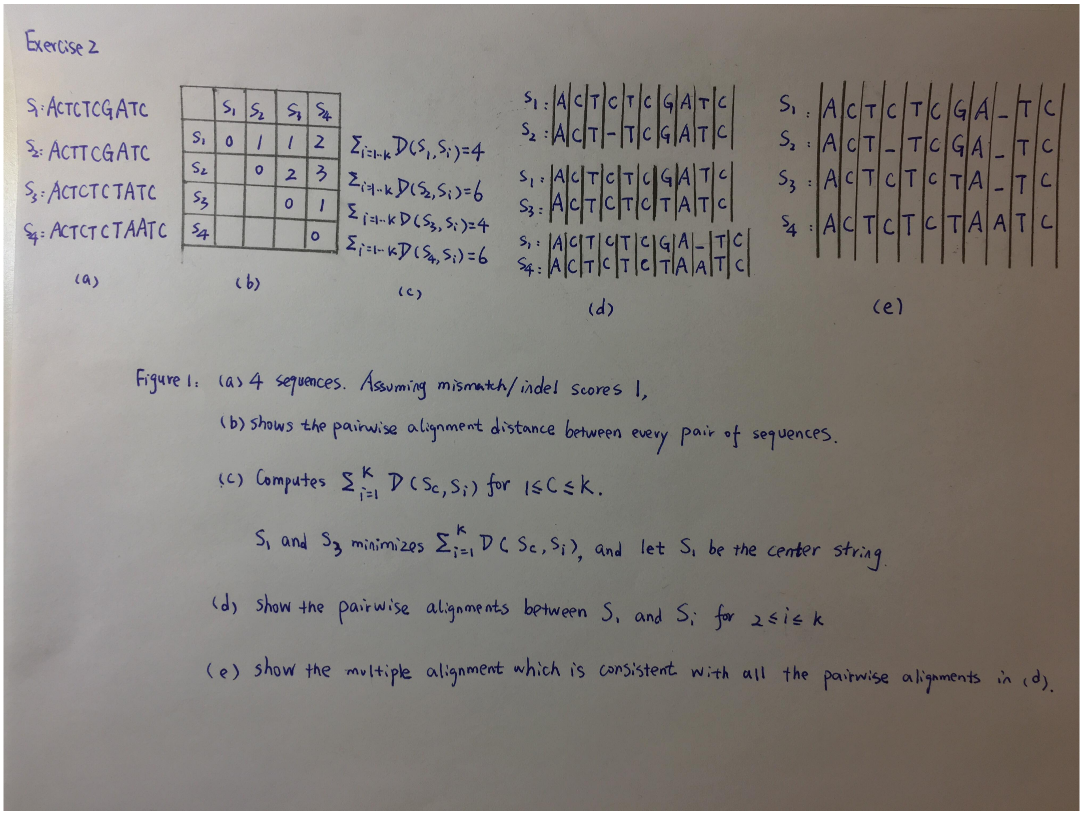
CSCI 551 HW4

Yin Yalan and John M. Singleton

due by 11:59 PM on T 11/5/2019

1 Problem 1 (Sung P6.2)



Exercise 2

S_1: ACTCTCGATC

S_2: ACTTCGATC

S_3: ACTCTCTATC

S_4: ACTCTCTAATC

(a)

	S_1	S_2	S_3	S_4
S_1	0	1	1	2
S_2		0	2	3
S_3			0	1
S_4				0

(b)

$\Sigma_{i=1..k} D(S_1, S_i) = 4$

$\Sigma_{i=1..k} D(S_2, S_i) = 6$

$\Sigma_{i=1..k} D(S_3, S_i) = 4$

$\Sigma_{i=1..k} D(S_4, S_i) = 6$

(c)

S_1 :	A	C	T	C	T	C	G	A	T	C
S_2 :	A	C	T	-	T	C	G	A	T	C

S_1 :	A	C	T	C	T	C	G	A	T	C
S_3 :	A	C	T	C	T	C	T	A	T	C

S_1 :	A	C	T	C	T	C	G	A	_	T	C
S_4 :	A	C	T	C	T	C	T	A	A	T	C

(d)

S_1 :	A	C	T	C	T	C	G	A	_	T	C
S_2 :	A	C	T	_	T	C	G	A	_	T	C
S_3 :	A	C	T	C	T	C	T	A	_	T	C
S_4 :	A	C	T	C	T	C	T	A	A	T	C

(e)

Figure 1: (a) 4 sequences. Assuming mismatch/indel scores 1,

(b) shows the pairwise alignment distance between every pair of sequences.

(c) Computes $\Sigma_{i=1}^{K} D(S_c, S_i)$ for $1 \le C \le K$.

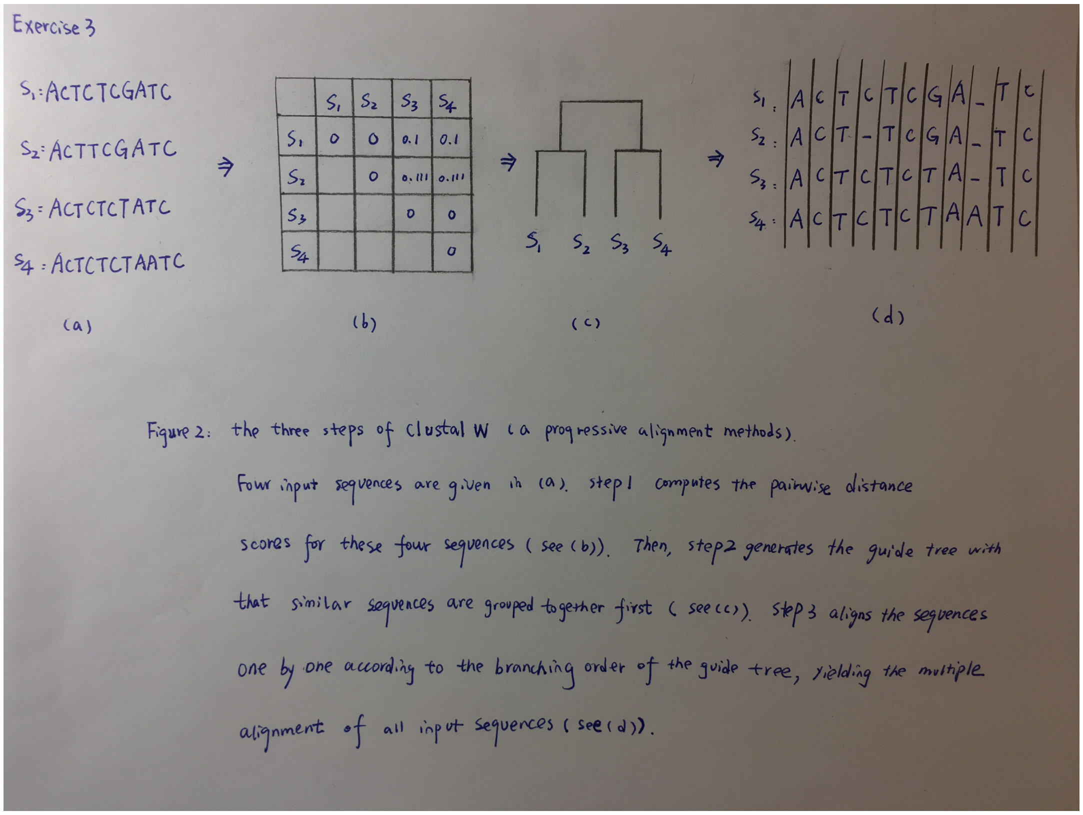
S_1 and S_3 minimizes $\Sigma_{i=1}^{k} D(S_c, S_i)$, and let S_1 be the center string.

(d) show the pairwise alignments between S_1 and S_i for $2 \le i \le k$

(e) show the multiple alignment which is consistent with all the pairwise alignments in (d).

2 Problem 2 (Sung P6.3)

Below is the steps for computing the multiple sequence alignment of the 4 sequences using ClustalW:



Exercise 3

S_1:ACTCTCGATC

S_2:ACTTCGATC

S_3:ACTCTCTATC

S_4:ACTCTCTAATC

(a)

	S_1	S_2	S_3	S_4
S_1	0	0	0.1	0.1
S_2		0	0.111	0.111
S_3			0	0
S_4				0

(b)

(c)

S_1 :	A	C	T	C	T	C	G	A	_	T	C
S_2 :	A	C	T	-	T	C	G	A	_	T	C
S_3 :	A	C	T	C	T	C	T	A	_	T	C
S_4 :	A	C	T	C	T	C	T	A	A	T	C

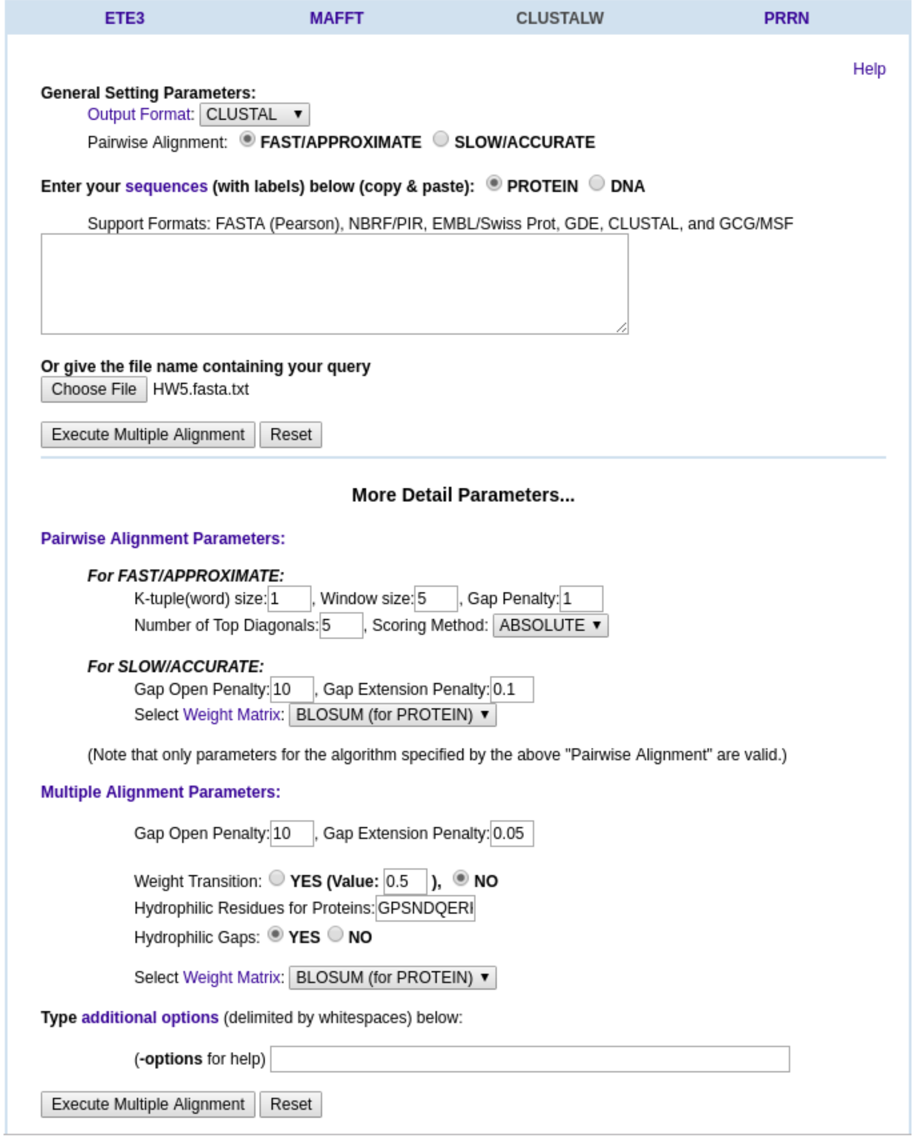
(d)

Figure 2: the three steps of Clustal W (a progressive alignment methods).

Four input sequences are given in (a). step 1 computes the pairwise distance scores for these four sequences (see (b)). Then, step2 generates the guide tree with that similar sequences are grouped together first (see (c)). Step 3 aligns the sequences one by one according to the branching order of the guide tree, yielding the multiple alignment of all input sequences (see (d)).

Below are the screenshots for running the CLUSTALW Servers:

1. the parameters set up as follows:



General Setting Parameters:

Output Format: CLUSTAL ▾

Pairwise Alignment: ● FAST/APPROXIMATE ○ SLOW/ACCURATE

Enter your sequences (with labels) below (copy & paste): ● PROTEIN ○ DNA

Support Formats: FASTA (Pearson), NBRF/PIR, EMBL/Swiss Prot, GDE, CLUSTAL, and GCG/MSF

Or give the file name containing your query

Choose File HW5.fasta.txt

Execute Multiple Alignment | Reset

More Detail Parameters...

Pairwise Alignment Parameters:

For FAST/APPROXIMATE:

K-tuple(word) size: 1 , Window size: 5 , Gap Penalty: 1

Number of Top Diagonals: 5 , Scoring Method: ABSOLUTE ▾

For SLOW/ACCURATE:

Gap Open Penalty: 10 , Gap Extension Penalty: 0.1

Select Weight Matrix: BLOSUM (for PROTEIN) ▾

(Note that only parameters for the algorithm specified by the above "Pairwise Alignment" are valid.)

Multiple Alignment Parameters:

Gap Open Penalty: 10 , Gap Extension Penalty: 0.05

Weight Transition: ○ **YES (Value:** 0.5 **),** ● **NO**

Hydrophilic Residues for Proteins: GPSNDQERI

Hydrophilic Gaps: ● **YES** ○ **NO**

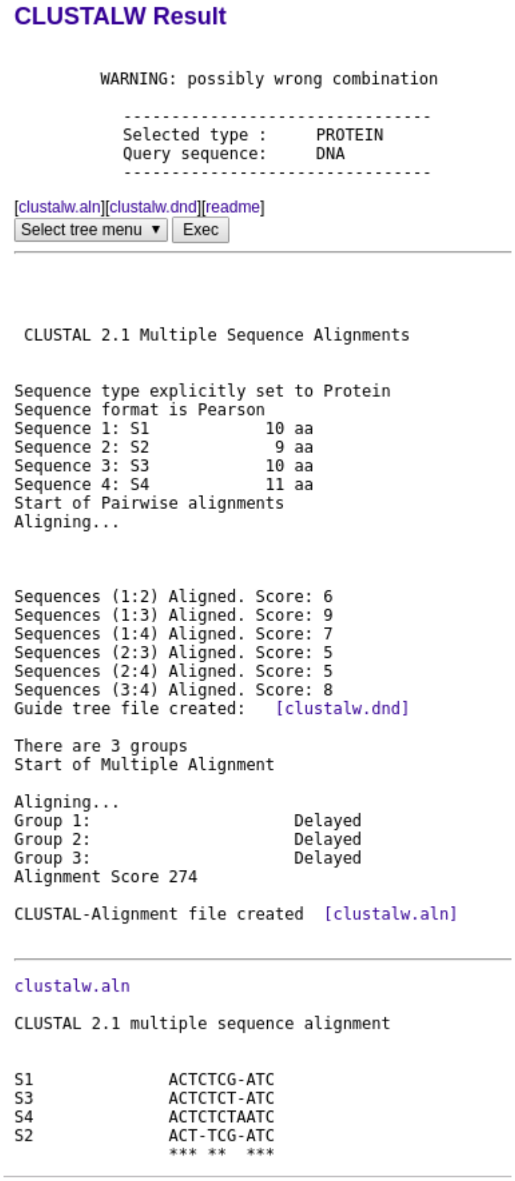
Select Weight Matrix: BLOSUM (for PROTEIN) ▾

Type additional options (delimited by whitespaces) below:

(**-options** for help)

Execute Multiple Alignment | Reset

2. the multiple sequence alignment output as follows:



CLUSTALW Result

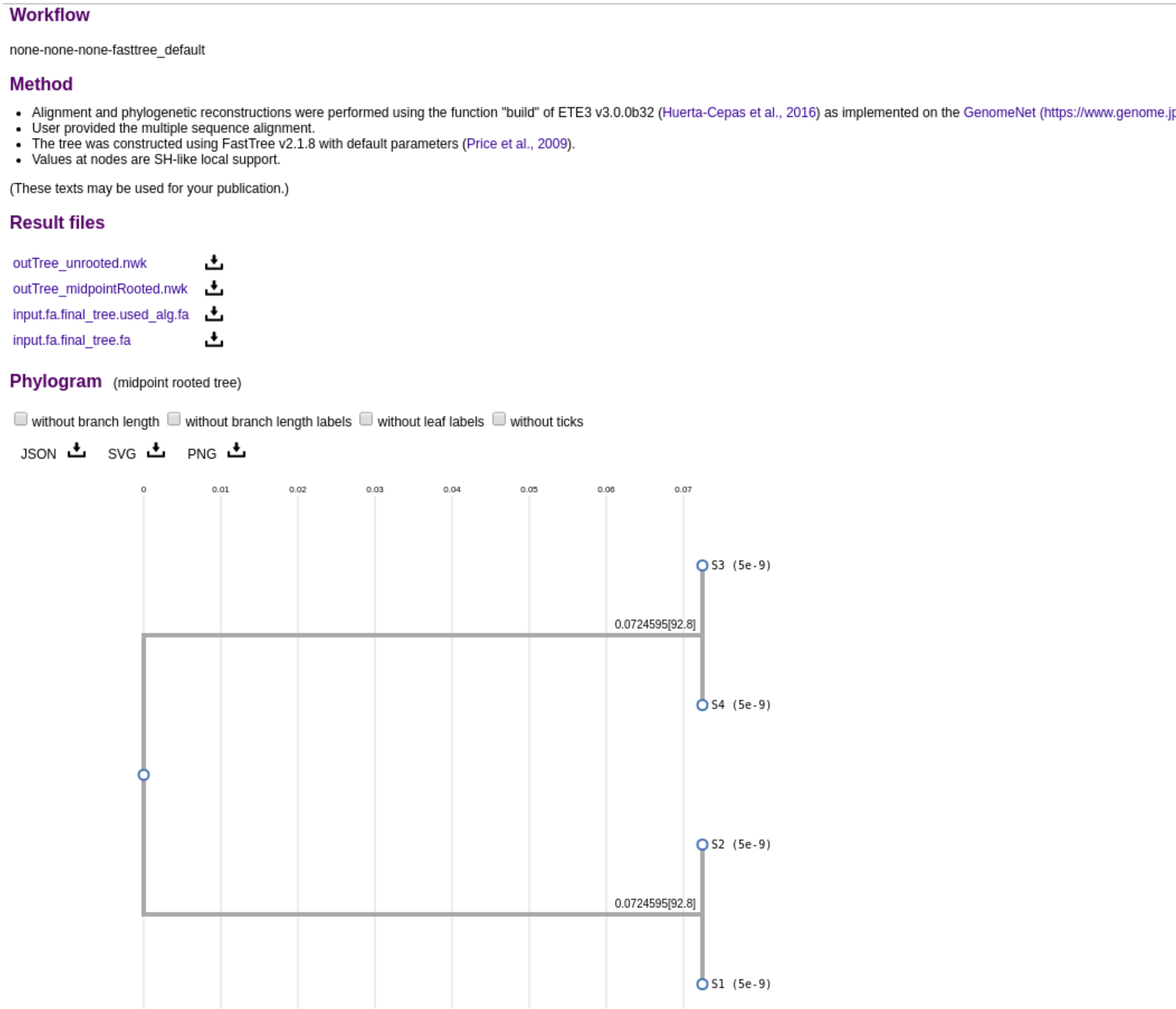
```
        WARNING: possibly wrong combination

           -------------------------------
           Selected type :     PROTEIN
           Query sequence:     DNA
           -------------------------------
```

[clustalw.aln][clustalw.dnd][readme]

```
 CLUSTAL 2.1 Multiple Sequence Alignments


Sequence type explicitly set to Protein
Sequence format is Pearson
Sequence 1: S1            10 aa
Sequence 2: S2             9 aa
Sequence 3: S3            10 aa
Sequence 4: S4            11 aa
Start of Pairwise alignments
Aligning...



Sequences (1:2) Aligned. Score: 6
Sequences (1:3) Aligned. Score: 9
Sequences (1:4) Aligned. Score: 7
Sequences (2:3) Aligned. Score: 5
Sequences (2:4) Aligned. Score: 5
Sequences (3:4) Aligned. Score: 8
Guide tree file created:   [clustalw.dnd]

There are 3 groups
Start of Multiple Alignment

Aligning...
Group 1:                     Delayed
Group 2:                     Delayed
Group 3:                     Delayed
Alignment Score 274

CLUSTAL-Alignment file created  [clustalw.aln]
```

```
clustalw.aln

CLUSTAL 2.1 multiple sequence alignment


S1              ACTCTCG-ATC
S3              ACTCTCT-ATC
S4              ACTCTCTAATC
S2              ACT-TCG-ATC
                *** **  ***
```

3. the fast-tree output as follows:



Workflow

none-none-none-fasttree_default

Method

- Alignment and phylogenetic reconstructions were performed using the function "build" of ETE3 v3.0.0b32 (Huerta-Cepas et al., 2016) as implemented on the GenomeNet (https://www.genome.jp
- User provided the multiple sequence alignment.
- The tree was constructed using FastTree v2.1.8 with default parameters (Price et al., 2009).
- Values at nodes are SH-like local support.

(These texts may be used for your publication.)

Result files

outTree_unrooted.nwk
outTree_midpointRooted.nwk
input.fa.final_tree.used_alg.fa
input.fa.final_tree.fa

Phylogram (midpoint rooted tree)

☐ without branch length ☐ without branch length labels ☐ without leaf labels ☐ without ticks

3 Problem 3 (Sung P6.4)

We can imagine a dynamic programming solution that is similar to the solution given in C6S4 for aligning two sequences. We will build up the solution by moving parallel to axis of the first sequence (of the hypercube-looking table/matrix). First we will fill in a line segment next to the characters of the first sequence (from the first character to the last), and next to the empty strings of the rest of the k sequences. Then we fill in a line segment next to the characters of the first sequence (from the first character to the last), and next to the first character of the second sequence and the empty strings of the rest of the k sequences. Etc.

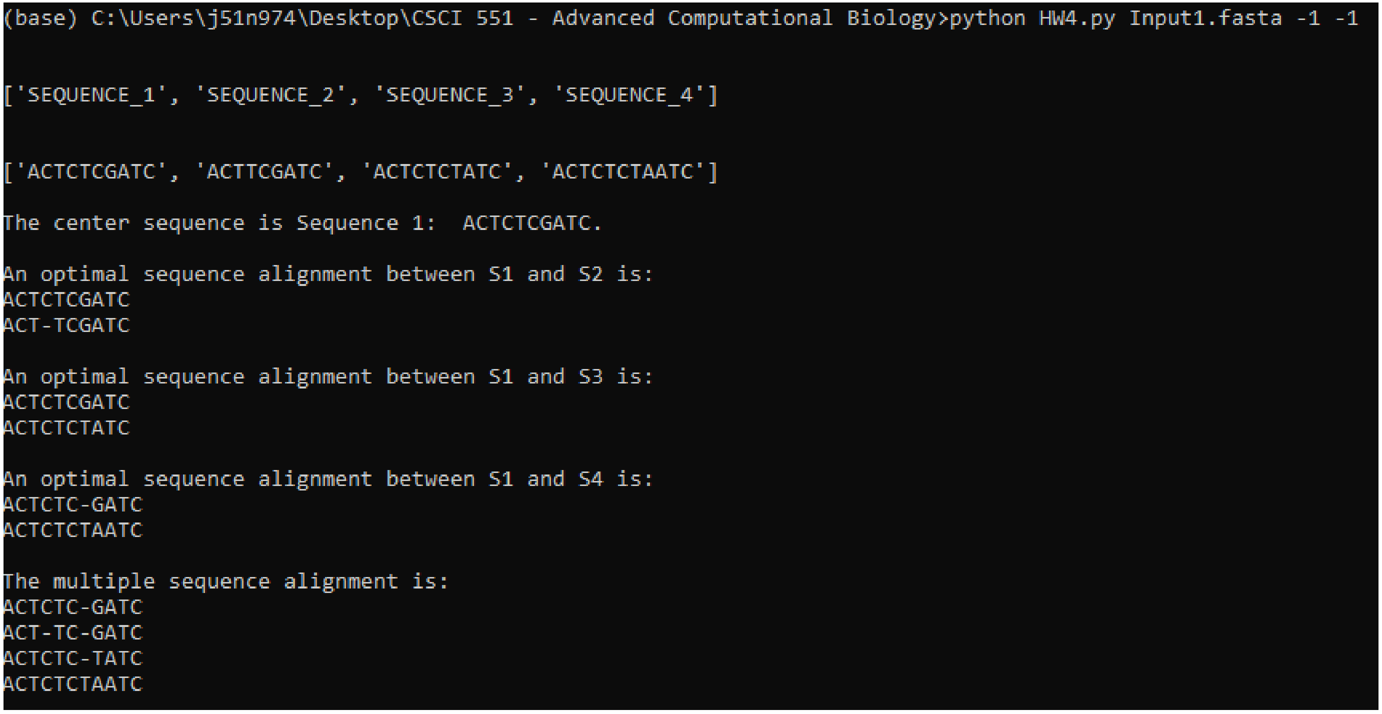
The recurrence relation is similar, but now the score in each new position is the maximum score of the $2^k - 1$ different cases (Think a 0 or 1 for each sequence, but all 0's is not allowed.) For example, this case could correspond to a mismatch between the current characters in the first and second sequences, and a deletion in the rest of the sequences.

If the lengths of the sequences S_1,... S_k are given by n_1,...n_k, respectively, then the running time is O($\Pi_{i=1}^{k} n_i$).

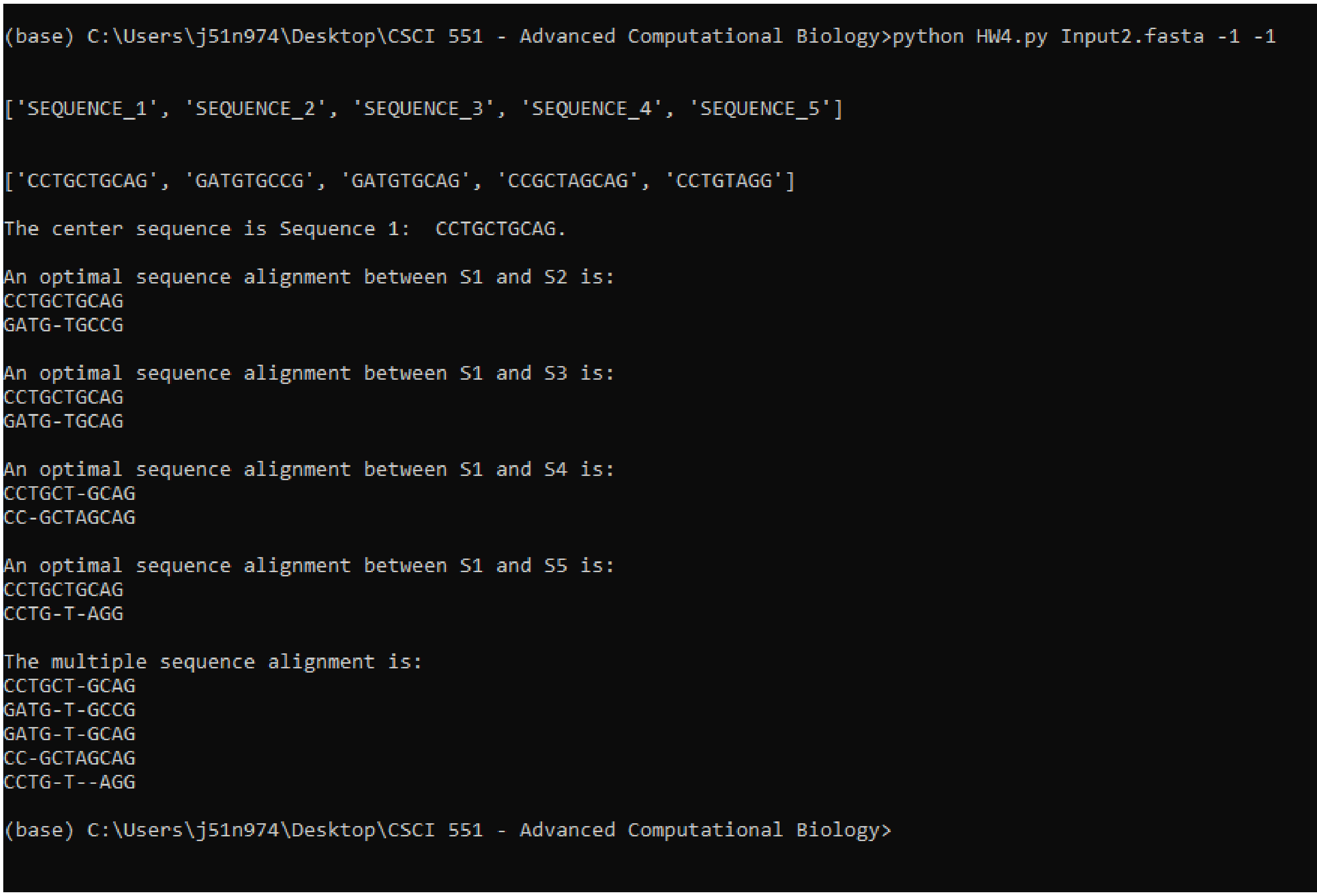
4 Problem 4 and Bonus

The first screenshot below (Input1.fasta) is for the set of sequences from Problem 1.

We implemented Steps 1-3 and part of Step 4; our implementation can find the multiple sequence alignment in the case where the center sequence happens to be the first sequence. Handling the case where the center sequence does not happen to be the first sequence is still on our TODO list.



```
(base) C:\Users\j51n974\Desktop\CSCI 551 - Advanced Computational Biology>python HW4.py Input1.fasta -1 -1


['SEQUENCE_1', 'SEQUENCE_2', 'SEQUENCE_3', 'SEQUENCE_4']


['ACTCTCGATC', 'ACTTCGATC', 'ACTCTCTATC', 'ACTCTCTAATC']

The center sequence is Sequence 1:  ACTCTCGATC.

An optimal sequence alignment between S1 and S2 is:
ACTCTCGATC
ACT-TCGATC

An optimal sequence alignment between S1 and S3 is:
ACTCTCGATC
ACTCTCTATC

An optimal sequence alignment between S1 and S4 is:
ACTCTC-GATC
ACTCTCTAATC

The multiple sequence alignment is:
ACTCTC-GATC
ACT-TC-GATC
ACTCTC-TATC
ACTCTCTAATC
```



```
(base) C:\Users\j51n974\Desktop\CSCI 551 - Advanced Computational Biology>python HW4.py Input2.fasta -1 -1


['SEQUENCE_1', 'SEQUENCE_2', 'SEQUENCE_3', 'SEQUENCE_4', 'SEQUENCE_5']


['CCTGCTGCAG', 'GATGTGCCG', 'GATGTGCAG', 'CCGCTAGCAG', 'CCTGTAGG']

The center sequence is Sequence 1:  CCTGCTGCAG.

An optimal sequence alignment between S1 and S2 is:
CCTGCTGCAG
GATG-TGCCG

An optimal sequence alignment between S1 and S3 is:
CCTGCTGCAG
GATG-TGCAG

An optimal sequence alignment between S1 and S4 is:
CCTGCT-GCAG
CC-GCTAGCAG

An optimal sequence alignment between S1 and S5 is:
CCTGCTGCAG
CCTG-T-AGG

The multiple sequence alignment is:
CCTGCT-GCAG
GATG-T-GCCG
GATG-T-GCAG
CC-GCTAGCAG
CCTG-T--AGG

(base) C:\Users\j51n974\Desktop\CSCI 551 - Advanced Computational Biology>
```