Bilkent University

CS481: Bioinformatics Algorithms

Homework Assignment #6

Fall 2020

INSTRUCTIONS

- Solve the following problems.
- You must write your code yourself. Sufficient evidence of plagiarism will be treated the same as for plagiarism or cheating.
- Non-compiling submissions will not be evaluated.
- Your code must be complete.
- Do not submit the program binary. You must submit the following items:
  - All of the source files
  - A script to compile the source code and produce the binary (`Makefile`).
  - A `README.txt` file that describes how the compilation process works.
- Submit your answers **ONLY** through the Moodle page.
- **Zip** your files and send them in only one zipped file. File name format `surname_name_hw#.zip`
- C / C++, Python 3, Java will be used as programming language. STL is allowed. The use of `getopt` function is **compulsory** for C/C++ programs. Python programs **MUST** use `argparse` module. Java programs **MUST** use an argument parser such as `ArgParser`
- All submissions will be compiled and tested on **Dijkstra server**.
- All submissions must be made strictly before the stipulated deadline.
- The overall fastest implementation wins. **Bonus** will be given for the fastest code.

1) GUIDE TREE CONSTRUCTION

Aim: Given **n** DNA sequences, $2 \leq n \leq 25$, in a FASTA file, you must implement a **Guide Tree Construction** using the **UPGMA algorithm**.

You may utilize your own **Affine gap** implementation of **Needleman-Wunsch** algorithm from previous assigments or you can write it from the scratch. You will take gap opening and extension penalties, match score, and mismatch penalty via parameters. The length of the input sequences might be at most **500** characters.

For the sake of simplicity, we give the alignment file (.fasta) for three sequences as an example described below. Output will be **in Newick tree format** (`http://evolution.genetics.washington.edu/phylip/newicktree.html`).

Create your distance matrix by first aligning with Needleman-Wunsch, and then calculating edit distances of the pairwise alignments. Do **NOT** give the intermediate steps as output.

Parameters:

- **−−fasta** FASTA-formatted file containing all sequences. This file may include up to 25 sequences. You may need to parse this file twice to count the number of sequences and then load them, if necessary.
- **−−gapopen** gap opening penalty
- **−−gapext** gap extension penalty
- **−−match** match score
- **−−mismatch** mismatch penalty

Output:

- **−−out:** sequences.tree in Newick format.

2) EXAMPLE

Command line examples: Be sure that your code works using the following command (just one line):

```
buildUPGMA --fasta seq.fasta \
           --match 5 \
           --mismatch -3 \
           --gapopen -8 \
           --gapext -1 \
           --out seq.tree
```

seq.fasta

```
>A
CTAGATAATTGCCAGATGATCAAATTTATAT
>B
CTAGATAATCATGCTAGCTAGTGCACAAATTTATAT
>C
CTAGATAATTGGAATGTCGATCGATCG
```

seq.tree

```
((A:4.5, B:4.5):2.75, C:7.25);
```