

Sequence Bioinformatics

WS 2025/26

Assignment 2

Due: Oct-29, 10 am

In this assignment, please implement a Java program `NW.YOUR_NAME.java` that reads as input a file containing two sequences in FastA format and performs optimal global alignment using the Needleman-Wunsch algorithm, writing the output to the console.

Please download the file `assignment02.zip` and implement your code in the provided file `NW.YOUR_NAME.java`. You will also find two input files. The file `small.fasta` provides two reasonably short sequences for running your code on.

The file `mtDNA.fasta` contains two long sequences, the full mtDNA reference genomes for human and mouse. Please use this file in task 3.

The command-line usage of your program should be as follows:

```
NW.YOUR_NAME <file> <mode> <match> <mismatch> <gap>
```

where

- `<file>` is a file that contains exactly two sequences in FastA format,
- `<mode>` is either `quadratic` for `linear` to run either the quadratic-space or linear-space versions of the algorithm, respectively,
- `<match>` is a positive integer match score,
- `<mismatch>` is a negative integer mismatch score, and
- `<gap>` is a positive integer (linear) gap penalty.

1 Needleman-Wunsch basic implementation (3 points)

When launched using the option `quadratic`, the program will run your own implementation of the Needleman-Wunsch algorithm and traceback, using the basic *quadratic-space* formulation.

The program should then print out the optimal score and an alignment that achieves the score.

Also print out the number of milliseconds used and the maximum amount of memory used to represent the DP matrix.

2 Needleman-Wunsch with linear space (4 points)

When launched using the option `linear`, the program will run your own implementation of the Needleman-Wunsch algorithm and traceback, using the *linear-space* Hirschberg divide-and-conquer approach.

The program should then print out the optimal score and an alignment that achieves the score.

Also print out the number of milliseconds used and the maximum amount of memory used to represent the DP matrix.

3 Performance comparison (2 points)

Based on the file `mtDNA.fasta`, produce subproblems of different lengths and using these to plot the time requirement and memory requirement for both modes, quadratic and linear, as a function of sequence length.

4 Revision (exam-style question) (1 point)

Write down and explain the main dynamic programming recursion for global alignment using affine gap penalties, involving three matrices M , I_x and I_y .