

Assembler

Due: Thursday, April 10th 2:30pm

1 Overview

Your job is to simulate the process of sampling random pieces of a long piece of DNA and assembling the fragments to reconstruct the original sequence. In the read generating phase, you will need to create a set of random pieces cut from a given DNA sequence, and then select a subset of these fragments for sequencing and assembly. Given these fragments, you will attempt to reconstruct the original sequence and compare your reconstruction with the original.

The real-world problem is very challenging because many factors and uncertainty affect the accuracy of the assembly process. In the scope of our class, we assume that all random pieces have the same orientation.

The following tasks can be used to assemble the random pieces into longer contigs. Note that it is not always possible to reconstruct the original DNA sequence.

Your job is to propose detailed algorithms for above steps and implement the genome assembly process. Your program should read in a file containing a DNA sequence and print the assembly to a file. Experiment with your program by changing the parameters and discuss your findings.

Please read this specification carefully.

2 Parameterization

Your program should accept a number of parameters on the command line. The TAs should be able to change these without modifying the source code of your program.

- **Number of Genomes:** The number of genomes that your program should fragment and sample from.
- **Sampling Rate:** The fraction of fragments that will be sampled and sequenced. This should be a real number in the range $[0, 1]$.
- **Error Rate:** This parameter represents the rate at which sequencing errors occur. You may assume that all sequencing errors are substitutions. This should be real number in the range $[0, 1]$.
- **Read Length:** Your program should accept a mean fragment size and standard deviation.

3 I/O Specification

Input

Read a genome from a FASTA file format. **Your program must also be able to accept a file containing newline delimited reads. You can specify a commandline option for this.**

Output

The READ GENERATOR should output the contigs to a file. They should be newline delimited. Your program should also be able to output the randomly generated reads to a file as newline delimited strings.