CMPE 59H: Bioinformatics, Fall 2017
Assignment 2 - Genome Assembly - Creating Contigs
Due: 04/01/2018, 23:00

Most assemblies have gaps in k-mer coverage, which causes the de Bruijn graph to have missing edges. In such cases, the search for an Eulerian path fails. In this case, contigs (i.e., long, contiguous segments of the genome) rather than entire chromosomes are assembled.

In this assignment you will implement a program that generates contigs from a collection of reads with possibly imperfect coverage. Contigs can be obtained from the maximal non-branching paths in the de Bruijn graph. A path in a graph is called non-branching if in(v) = out(v) = 1 for each intermediate node v of this path, i.e., for each node except possibly the starting and ending node of a path. A maximal non-branching path is a non-branching path that cannot be extended into a longer non-branching path. The input to your program will be a plain text file containing a list of k-mers (reads). Your program should output all contigs in the de Bruijn graph constructed from these reads. Note that your program should first construct the *de Bruijn graph* for the input collection of k-mers and then find the maximal non-branching paths on this graph.

Test your program using the provided sample-input.txt and test-input.txt files. In your report, include *screenshots* from running your program on these files and the results produced.

Submission: Please submit your assignment using Moodle. Upload a single zip file named as YourNameSurname.zip. Your zip file should include your report, your source code, and the corresponding readme file. You can use any programming language of your choice. But, your readme file should clearly explain how to run your program.

Late Submission: You are allowed a total of 3 late days on homeworks with no late penalties applied. You can use these 3 days as you wish. For example, you can submit the first homework 2 days late, then the second homework 1 day late. After using these 3 extra days, 10 points will be deducted for each late day.