605.620: Algorithms for Bioinformatics

William McElhenney

wmcelhe1

Project 1 Analysis

September 24, 2019

September 25, 2019

**Project 1 Analysis**

Analysis of Ordinary Matrix Multiplication

The ordinary matrix multiplication algorithm can be implemented as a set of three nested loops to iterate through the matrix and add the correct products to an initialized output matrix. For a square matrix of dimension *n* the time complexity is O(n3) as we need iterate n3 times to hit all the values to add to the output. This also means that we must make *n3*multiplications meaning line 81 will execute exactly n3 times for a given matrix).

An implementation for ordinary matrix multiplication can be seen in *matrix\_tools.py* at   
lines 57 – 83. As can be seen, it is as I described it above, a set of three nested loops all going from 0 to *n*.

Analysis of Strassen Matrix Multiplication

Strassen’s algorithm cleverly implements a recursive matrix multiplication algorithm by splitting matrices into quarters, defining specific addition/subtraction operations on those partitions, and then recursively multiplying the sums/differences from those operations. The class text states that this algorithm has time complexity.

By breaking the algorithm down this algorithm completes