Weiming Raymond Luo
November 29, 2016

CS 405: Algorithm Analysis II
Homework 4: Polygon Decomposition

For this assignment, I implmented a polygon decomposition by creating non-intersecting chords that forms a triangle to obtain the minimum sum of triangle perimeters. When taking in the input text file of *polygon2.txt*, I achieved $\Theta(n^3)$ asymptotic complexity by creating an $n \times n$ matrix, where n is the amount of vertex in the input text file. The matrix is filled by using the recursive matrix multiplication method of,

$$m[i,j] = \begin{cases} 0, & perimeter \geq m[i][j] \\ min_{i \leq k \leq j} m[i,k] + m[k,j] + perimeter(i,k,j), & perimeter < m[i,j] \end{cases}$$

This will fill the matrix diagonally with the updated minimum perimeter for the triangle found in the polygon. After filling the whole matrix, since the calculations is going diagonally, this means the final result will be on the matrix cell $m[0,n]$. When running the progrma using the *polygon2.txt*, the resulting total minimum triangle perimeter is 8328.604014137758. With the respective chords of the polygon,

CHORDS	
0	2
0	23
16	19
17	19
2	23
2	5
3	5
19	23
19	21
5	23
5	7
5	9
5	13
21	23
7	9
9	13
9	11
11	13
13	23
13	15
13	16
13	19

We can determine the asymptotic complexity to be $\Theta(n^3)$ as the matrix is iteratively filled in a diagonal by checking each possible non-intersecting chord for the decomposition. The iterative process will go through the matrix once but will check each side of the triangle for various combinations of chords to find the minimum perimeter. This process of checking each side of the triangle for combinations will take at most $\Theta(n^3)$.