Project 2 Executive Summary—Joseph Petitti

Brute Force Algorithm time efficiency:

$$T_B(n) = \sum_{i=0}^{n-1} \sum_{j=i+1}^{n} c = \frac{n * (n+1)}{2} = O(n^2)$$

Recursive Algorithm time efficiency:

$$T_R(n) = \begin{cases} T_B(n) : n \leq 3 \\ 2 * T_R\left(\frac{n}{2}\right) + n \log n + n : n > 3 \end{cases}$$

Master Method, Case 2: $n^{\log_2 2} < n \log n + n$

$$T_R(n) = O(n\,(\log n)^2)$$

Test Cases:

	Runtimes (seconds)			
n	**Recursive**	**Brute Force**	**Answer**	**Notes**
4	4.2667e-5	1.0666e-5	45.122	See “Test 1”
12	8.2774e-5	6.5279e-5	2.828	See “Test 2”
1000	2.6222e-2	4.6988e-1	14.036	1000 random points (see “Test 3”)
4	2.8587e-5	7.6800e-6	0.0	Four identical points (see “Test 4”)
5000	6.5678e-1	1.2942e+1	2.0	5,000 random points (see “Test 5”)

The exact points tested are listed in “readme.txt”

These tests show experimentally what the time efficiency equations above showed theoretically: Except in the case of very small *n*, the recursive algorithm is much more time efficient.