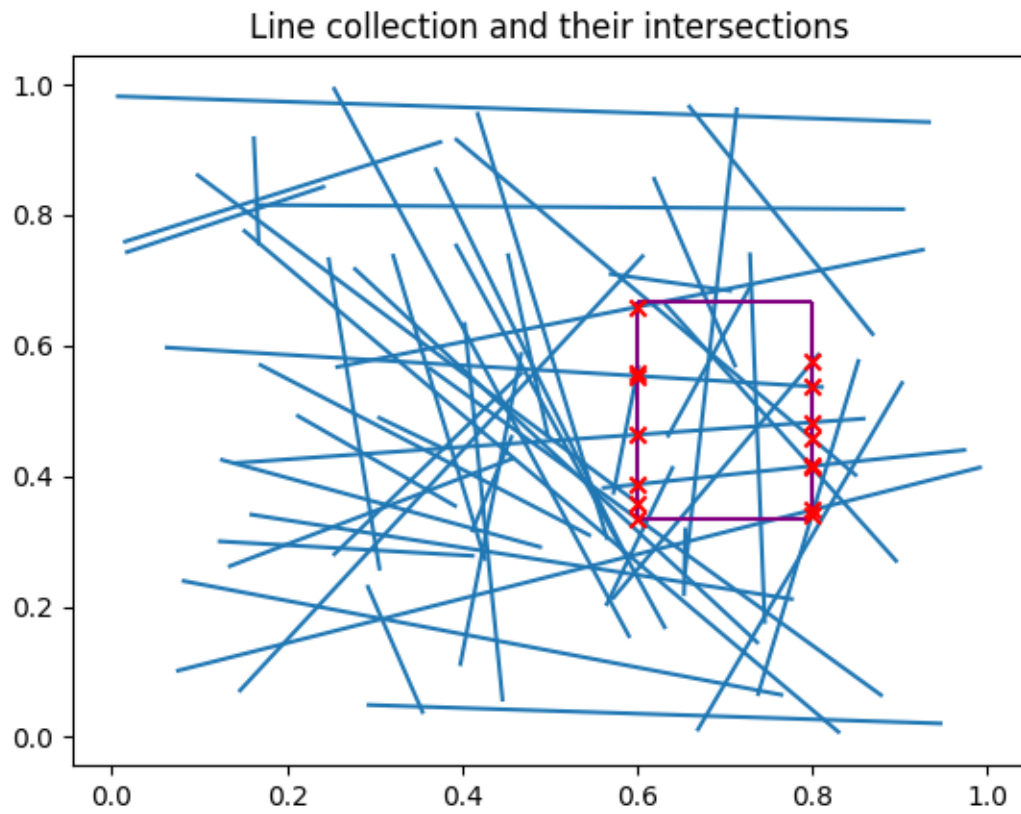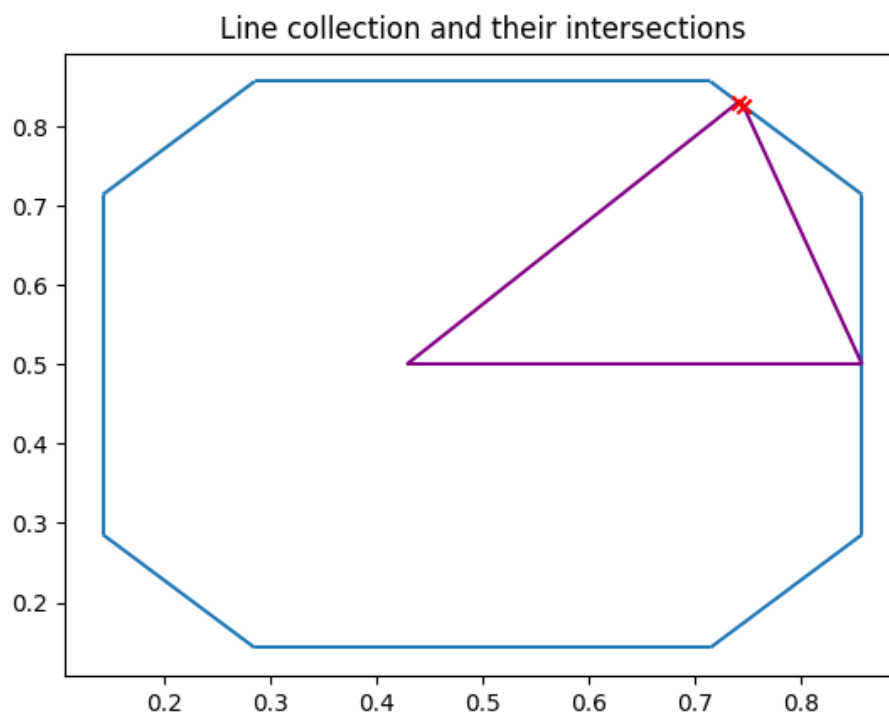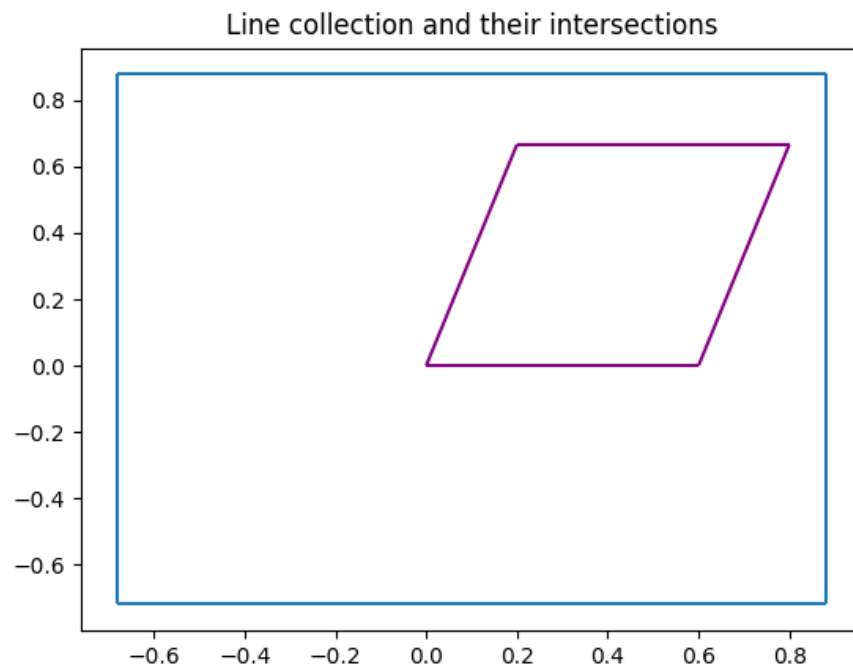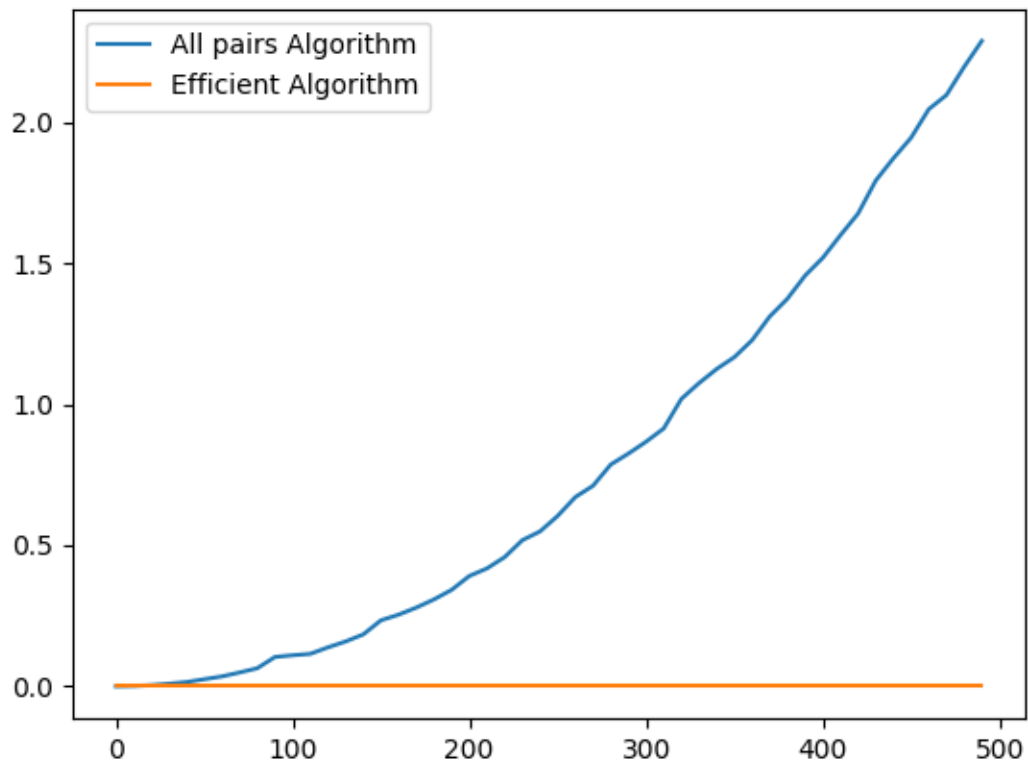- This is the output generated by the main.py function after writing the segment_pair_intersection() is written.

Line collection and their intersections

Line collection and their intersections

Line collection and their intersections

- After this the main.py generates an efficiency graph compared to the original algorithm which is shown below:

Time complexity

Let's examine each section of the above code separately in order to determine its temporal complexity:

- Since the line_eq function only does a given number of arithmetic operations, its constant time complexity is O(1).
- Additionally, the segment_pair_intersection function has an O(1) constant time complexity. It calculates the intersection location between two line segments using a given number of operations.
- The empty list that the efficient_intersections function currently returns has an O(1) constant time complexity.
- The time complexity of the all_pairs_intersections function, where "n" is the total number of line segments, is $O(n^2)$. It calls the segment_pair_intersection function, which has a constant time complexity, after iterating through each pair of line segments. As a result, $O(n^2)$ is the general time complexity.