Edge intersection test checked for 4 cross products. Ensuring the 2 points of the other edge are on diff sides of the first edge. This is checked the other way around too, this tells if they will intersect.

Edge intersection point is found out by using the parametric form of the line and see where they intersect. The parametric variable should be in the range 0 to 1 so that it is on the edge as otherwise the line can go to infinity.

AABB intersection test is quick and fast as only 4 comparisons are done to estimate if they may intersect or not. We need to do further check to ensure and confirm if they intersect.

Shamos is implemented with queue, intersection and multiple cases to find the intersections. It was a lot more complicated to implement in code than on paper.

I did all the implementation based on the sweep optimization and hence have a queue to run the loops.

We test 10, 100 and 1000 line segments and aim to get better performance for more edges. It prints the comparison with the naive method and tells much better or worse it is.

The line segment version 1 is very basic which has a limit to line segment length and has angle ranging from 0 to 360 degrees. For this set we get AABB to perform better by 75% for 1000 edges as compared to the naive method. Shamos gives better and worse performance based on the number of line segments.

The line segment version 2 I have put lines which are very short and hence will not intersect as much, thus we should get very high performance and the intersection point will not be checked, thus saving time. I get 90% improvement on this set of 1000 edges. For this set shamos shows improvements between 20-80%, but AABB still performs better than it for this case.

The line segment version 3 i have put lines which are either more horizontal or more vertical. Here we will have more perpendicular lines, which are more aligned to the axes we use. Hence the performance falls slightly as compared to the first version of segments. It is 67% for 1000 edges. Shamos shows some improvement over naive, but not more than AABB. Infact performs worse for more lines in this version.

Shamos implementation shows a lot more variation in performance based on how the line segments are. AABB is more consistent.