**Convex Hull Algorithm comparison and analysis**

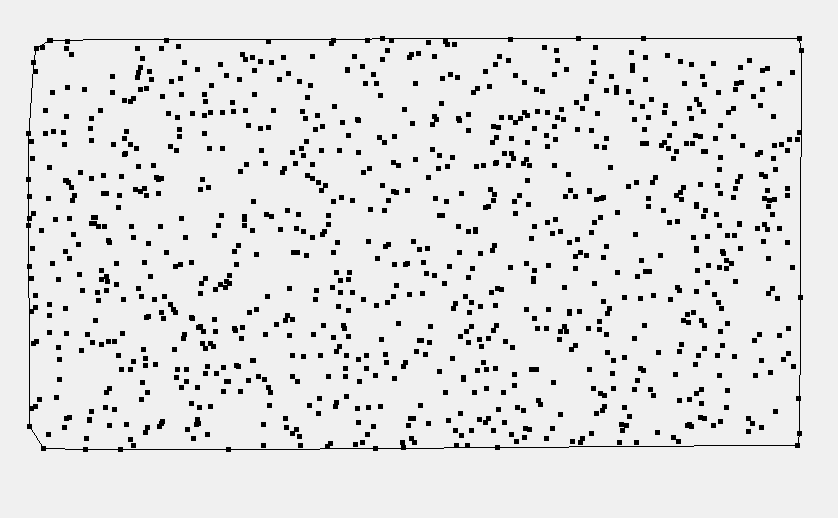
Algorithms discussed in this assignment include various algorithms to compute convex hull, which are Jarvis algorithm, Andrew’s algorithm and Graham’s scan. Out of these algorithms Jarvis algorithm is output dependent with the complexity of O(n\*h) where n is the number of input points and h is the number of points in convex hull. While the other two algorithms are of the order of O(n\*logn).

The practical analysis of these algorithms are being done in this assignment and the following table further helps to analyse these algorithms.

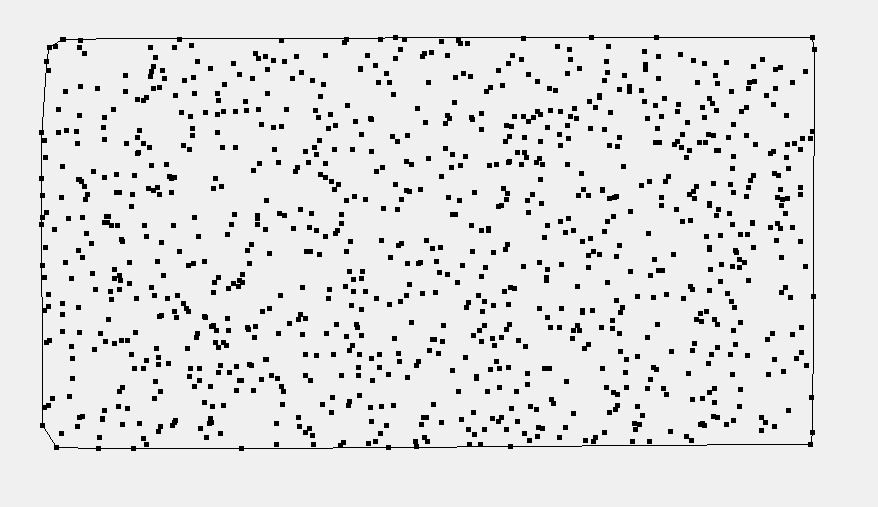
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Number of Points(n) | Points in Convex Hull(h) | Jarvis | Andrew | Graham’s Scan |
| 40 | 8 | ~0 | ~0 | ~0 |
| 1000 | 18 | 0.013 | 0.012 | 0.011 |
| 1000 | 914 | 0.065 | 0.014 | 0.016 |
| 10000 | 27 | 0.111 | 0.074 | 0.068 |
| 10000 | 5961 | 2.22 | 0.116 | 0.147 |
| 10000 | 4 | 0.095 | 0.066 | 0.063 |

This table contains the data for which there is enough variations between n and h so that the comparison between the output sensitive Jarvis algorithm and other two algorithms could be done.

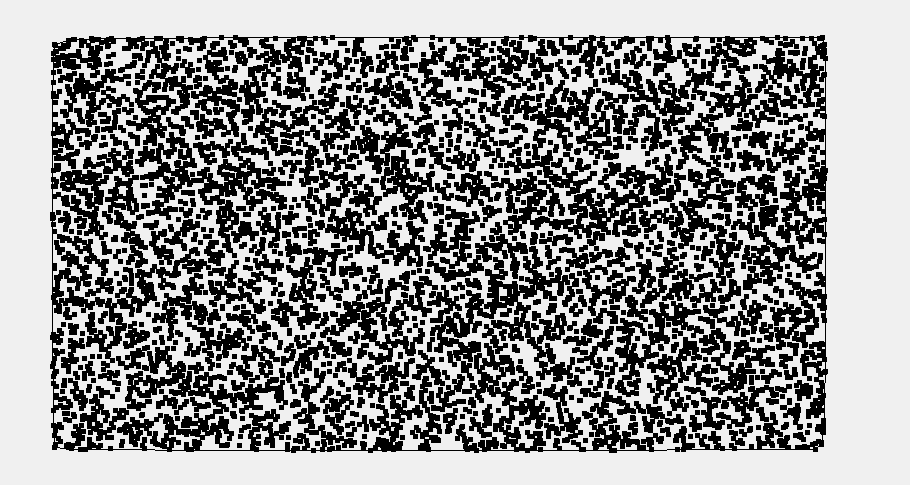
For the same n of 10000, Andrew algorithm and Graham’s Scan have a change of about 0.03s while the change in time is a little more than 2s in the case of Jarvis algorithm this shows us the output sensitive nature of the Jarvis algorithm.



Output in visualizer for Jarvis algorithm for 10^3 ordered data set (0.013s)



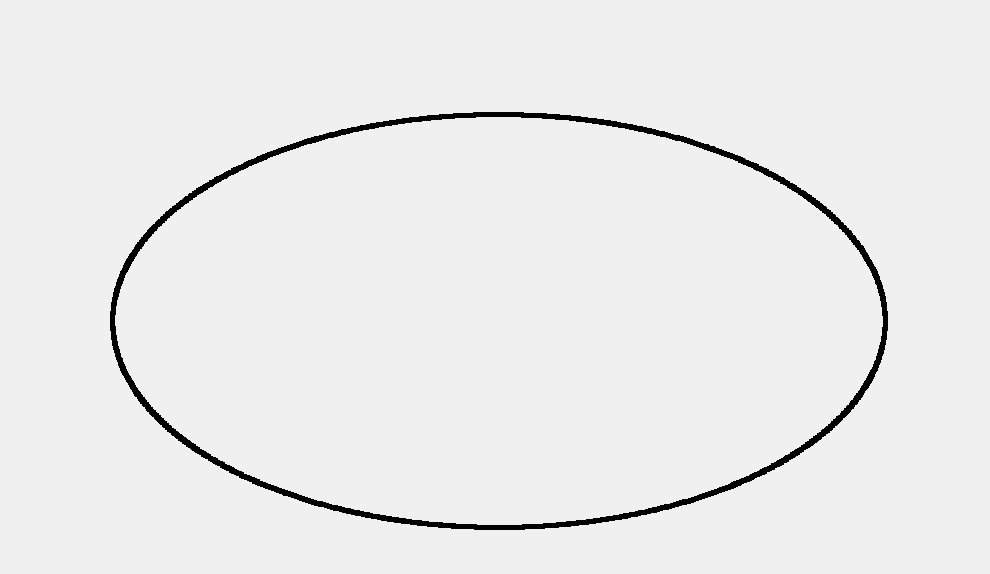
Output in visualizer for Andrew’s algorithm for 10^3 ordered data set (0.012s)



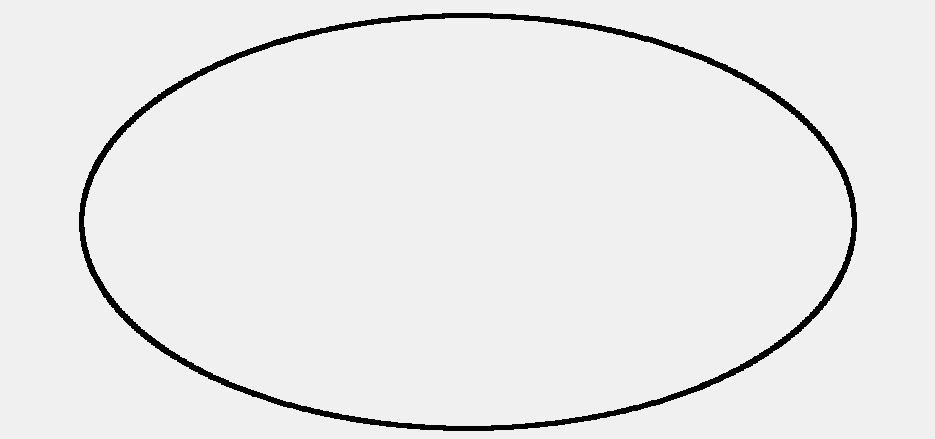
Output in visualizer for Jarvis algorithm for 10^4 ordered data set (0.113s)



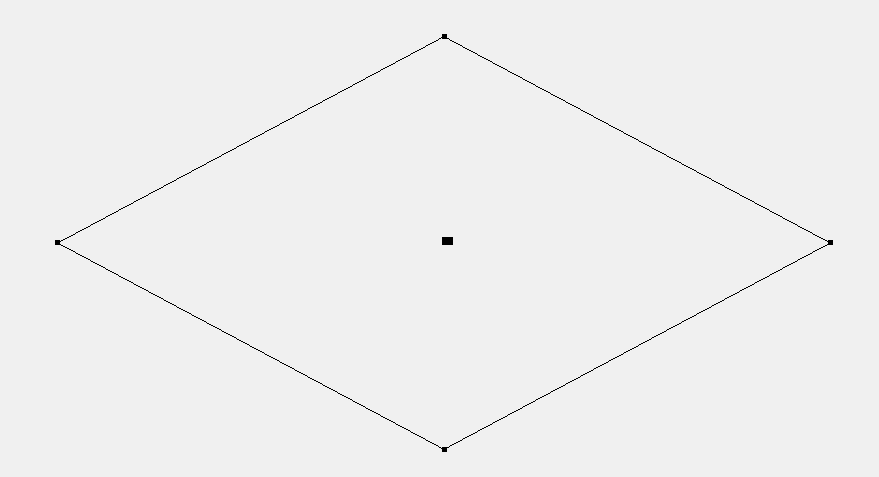
Output in visualizer for Andrew’s algorithm for 10^4 ordered data set (0.078s)



Output in visualizer for Jarvis algorithm for 10^4 ordered data set (2.221s)



Output in visualizer for Andrew’s algorithm for 10^4 ordered data set (0.148s)



Output in visualizer for Jarvis algorithm for 10^4 ordered data set (0.095s)

This example shows the real output sensitivity of Jarvis algorithm, the big black dot in middle is a cluster of randomly generated 10000 points while the other 4 points are away from the cluster, Jarvis algorithm gives an output in 10ms for this example but for the same ordered data set with 5000 points in convex hull it takes about 2000ms.

Thus it can be concluded that Jarvis Algorithm actually shows the output sensitivity while the performance of other two algorithms is kind of equivalent.

The data set used in these experiments was float which made the computations much costlier than the simple integer data.

Degenerate issues have been handled accordingly and those included the cases when the polygons couldn’t be formed; when the points are less than three or the case when all the points are collinear. Those issues have been handled in the function which tells whether the points are clockwise, anti-clockwise or collinear.