

Andrew's Algorithm

For this exercise the Andrew's Algorithm also known as the monotone chain algorithm used for the creation of convex hulls was implemented and the execution time was measured.

The data used for the creation of the hull was generated using the STL random library and a uniform distribution generating integer values in between 10 and 490 for the x and the y coordinate. The seed can be set with the "--seed" parameter, although it has a default value of 1234. The amount of points to be generated can also be set with the "--pointCount" parameter, but has a default value of 10. The program can also animate the steps it made to generate the hull if the "--graphic" parameter is added.

To obtain the time measurements the STL crono library with the high_resolution_clock function was used.

Algorithm:

The algorithm takes in a list of points. For it to work there have to be at least 3 points present in the list. The given points are then sorted by their x-coordinate and in case of a tie the y coordinate is used to determine the sorting order. An additional list is created where the points belonging to the final hull are saved. While the newly created list contains at least two points, and the sequence of the last two points and the next point from the original list do not make counter-clockwise turn the last point from the new list is removed. If this is not the case, the point from the original list is added to the new list.

This procedure is used for the creation of the lower hull as well as the upper hull. After all the points have been found, the list gets resized and the results are listed in counter-clockwise order

This algorithm has a time complexity of $O(n \log n)$ because it consists of two phases, the sorting of the points, and the making of the hull. The making of the hull has a $O(n)$ time complexity, because every point will be visited one time. The sorting part takes $O(n \log n)$, and because this part has a larger time complexity, then the whole algorithm has this time complexity.

This algorithm will work exactly the same no matter the arrangement of the data it is provided, because as explained, it sorts and goes through all the points.

Time measurement:

The tests were made with the default seed of 1234.

Amount of Data	Time (ms)
10	0.003
100	0.013
1000	0.089
10000	1.036
100000	11.231
1000000	124.608
10000000	1203.730
100000000	29262.800

