

Computing the Convex Hull of a set of points using Graham's Scan Algorithm

Leonard Han

May 2020

1 Introduction and how I handled the problem

The problem is pretty straight forward, as the algorithm is already established, and only the implementation is up to me. This makes the process really easy.

As an implementation, I've chosen to use standard c++ classes like vector and stack, and functions like sort, in order to shorten the amount of code that I need to write. I've stored the read points into a vector to make it easy to access and move them around, while the points seen as viable are stored into a stack.

2 Other possible solutions

This could be implemented in a bunch of different ways, by changing the data structure that stores the points, which would impact only the way you need to interact with it. The current vector/queue seemed to me the most convenient and time effective, which is why I've chosen it.

3 Examples

The example provided in the task has the wrong output, since one of the points was on the outer line drawn by another two, so here is the same example with the corrected output.

For the input: (2, -2), (4, -1), (4, 1), (3, 0), (2, 2), (1, 0), (0, 3), (0, 1)

We obtain the output: (0, 1), (0, 3), (4, 1), (4, -1), (2, -2)

4 Conclusion

In conclusion, the problem is in just choosing the data structure with which the algorithm is to be implemented. As a result of this, it wasn't complicated to make.