CSCI 3250 Project 2

Rafael Almeida

October 3, 2024

1 Handling Degeneracies

In Graham's scan algorithm, degenerate inputs include one-point and collinear points to solve the convex hull problem. My algorithm handles the degenerate inputs by having an if-statement to catch and return a single point. My code can handle two and three points, which may be considered degenerate outputs but do not cause any issues given my implementation. Moreover, my algorithm handles collinear points using the $left\_strictly$ function and always adds a point to the hull. In the context of Graham's scan algorithm, when my implementation finds two consecutive collinear points since the second point is not strictly left of the previously found segment, the first point in the collinear segment will get popped from the stack holding the hull. Then, since I always add a point at the end of every iteration that builds the hull, the last point in the collinear segment will be added. Essentially, this approaches always prefers the last point any collinear segment. I also ensure that the first point added to the hull (bottom-rightmost) is never removed.

2 Time Spent and Reflection

- < 20 minutes thinking about Graham's scan

- < 30 minutes implementing and testing Graham's scan

- > 2 hours thinking about my initializers and implementing them

- < 15 minutes documenting

- ≈ 3 hours total

This project was relatively easy. I learned most about creating points to form a specific figure but not much more about Graham's scan, C++, the language, or OpenGL. I wish we had used more of what we learned in class to create a project with some application. An example could be using convex hulls to detect the shortest path around a concave polygon.
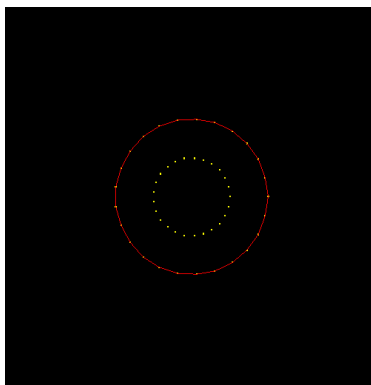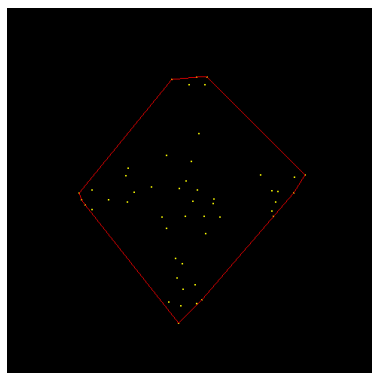
3   Initializers Used
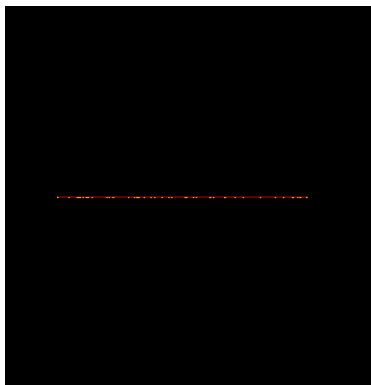

Figure 1: Circle

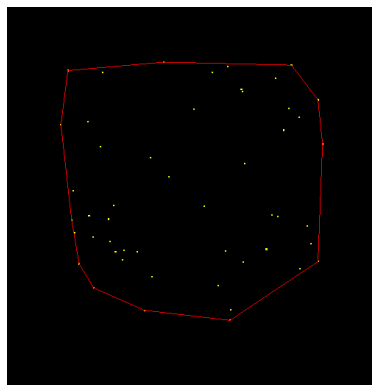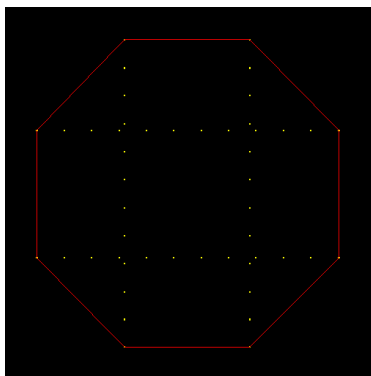Figure 2: Cross
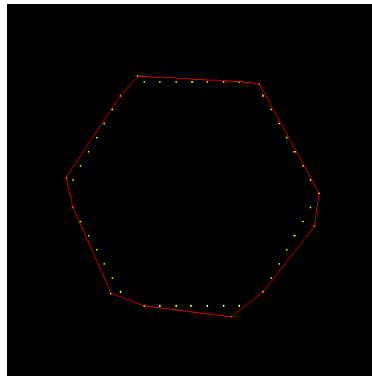

Figure 3: Line

Figure 4: Random

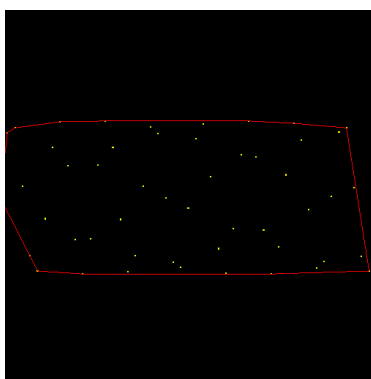Figure 5: Tic Tac Toe

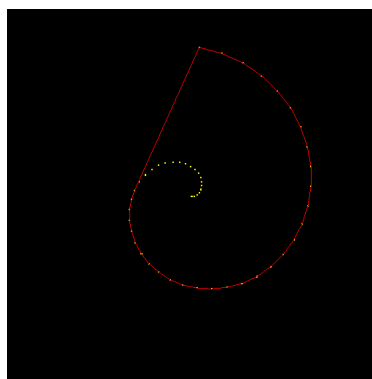

Figure 6: Honeycomb

Figure 7: Wave by Jack Kinsella
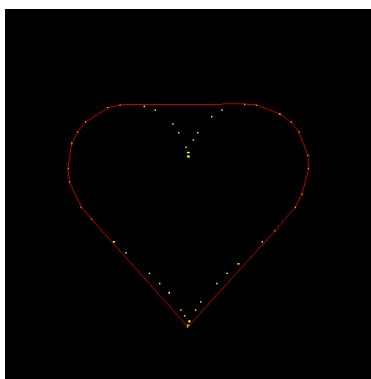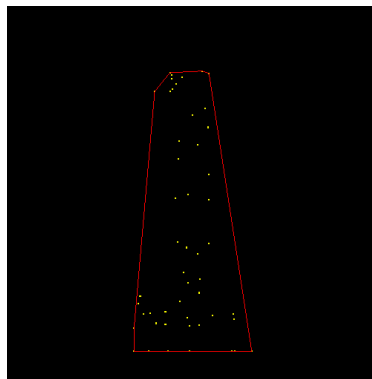
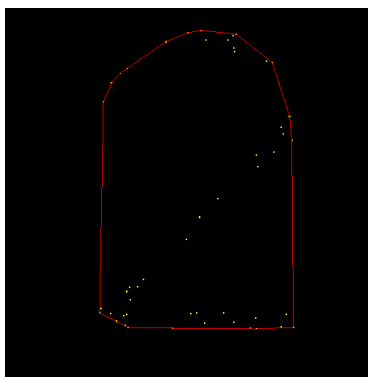

Figure 8: Spiral David Gelinas

Figure 9: Heart by Leah Dichter

Figure 10: One by Tom Han

Figure 11: Two by Tom Han