COT 4521-001: Introduction to Computational Geometry (Fall 2018)

Final Project

1 Objectives

For this assignment is the you will propose, implement, experiment or demonstrate, and report results for one or more algorithms. The goal is that you provide some new information to the rest of the class.

2 Ground Rules

This assignment is intended to be done alone. You may ask others for help with figuring out strategies. However, the code and reporting must be yours.

3 Assignment Instructions

3 deadlines will appear in Canvas. (1) Project proposal; (2) Mid-Project Report; (3) Final project submission. You will need to submit/upload the respective text, code, images, etc. for each.

- **Project Proposal** – The project proposal will be 2 paragraphs describing your planned project. The first paragraph will describe the topic. Be explicit—for example, if you are implementing 3 algorithms, tell me which 3 and why you picked them. The second paragraph should describe what experimentation or demonstration you plan to perform. Again, be explicit—if you plan performance comparison, describe what types of comparisons you will do. If you plan to demonstrate something, describe exactly how (like what types of input data you will use).

- **Mid-Project Report** – By the time the mid-project report is due, you should be done or close to done with your code. For this report, you will need to provide a copy of your code with instructions for compiling/running and a 2 paragraph status report. The first paragraph will provide detail about the status of the development and experimentation/demonstration. The second paragraph will describe any changes in the scope of your project.

- **Final Project Submission** – Your final submission will include your final code and a PowerPoint presentation given to the class. For the presentation, you will be allotted 8 minutes in class. You should produce a presentation of approximately 8 slides. In the slides, you will introduce the project, describe any experimentation, and discuss results and conclusions.

4 Potential Topics

The choice of topic is yours. Below is a suggested but not exclusive topic list. Important note: We can't have topics that overlap too much. If this is discovered, students with overlapping topics will be asked to revise their planned projects.

- Implement three algorithms (or fewer, if non-trivial) for a specific topic (triangulation; convex hulls; point searches; etc.).

- Implement one (non-trivial) algorithm in 3 or more programming languages (Python, Java, and C for example). The programming languages should have significant enough distinctions (interpreted vs GC/JIT compiled vs native compiled) that will potentially result in performance differences.
- Implement incremental Voronoi construction algorithm.
- Calculate geodesic distance using a relative neighborhood graph and compare to Euclidean distance.
- Multiple (> 2) convex polygon intersections, unions, differences, etc.
- Write a Processing sketch that animates any three (or fewer, if non-trivial) algorithms in your book.
- Find the intersection of halfplanes.
- Form a robust distance library (point-point, point-line, point-polygon, line-polygon, etc.).
- Implement the art gallery problem solution discussed in class. Enable checking if user selected guard locations cover the polygon.
- Compute and Voronoi diagram and Delaunay triangulation of a point set.
- Other ideas are welcome (see http://www.cse.usf.edu/~sarkar/cGeom/Computational_Geometry/Project.html for more examples)