# Project Proposal

Geometric Computing

Logan Weber
loganweb@mit.edu

The Lean proof assistant and programming language has made waves in the math community in recent years as an attractive platform for formalizing mathematics. Consequently, many undergraduate-level theorems in math have been formalized in Lean, and they have been collected into a single library, Mathlib.

This project will be a case study in formally verifying geometric algorithms in Lean, and we will leverage theorems in Mathlib as much as possible. Formal verification is considered by many to be an arduous process, and we aim to challenge this assumption and assess the level of abstraction that proofs can be expressed at in a modern theorem proving environment.

Concretely, we seek to implement and formally verify the following algorithms:

- The gridding-based algorithm for closest pair with help
- The gridding-based algorithm for finding helper $t$ for closest pair
- **(Stretch Goal)** Recursive, incremental LP solver

The Mathlib modules we suspect to find use for are:

- Hash Map
- Rationals
- Euclidean Geometry
- Metric Space
- Convex Analysis
- Normed Groups
- Normed Spaces
- $\ell_p$ Space

In order to compute with the functions we define, we cannot use $\mathbb{R}$ to represent points, since Lean does not provide a data type for computable approximations of the reals. Instead, we use $\mathbb{Q}$ as our carrier type, and we survey the challenges in doing so. For example, $\|\cdot\|_2$ computes the square root, which takes, e.g., $2 \in \mathbb{Q}$ to $\sqrt{2} \in \mathbb{R}$, so we can't use $\|\cdot\|_2$ as a norm. Our writeup will include commentary on the concessions of this flavor that are forced by modern proof assistants.