15-456/852 Computational Geometry, Fall 2017
Miller and Chu

Homework # 0 **Due: Wednesday, September 13**

Guidelines: Please justify all answers in a succienct (yet complete) manner. In particular, when presenting an algorithm the code if any should be presented at a high level. A full algorithm will contain the input, the output, and any loop invariants.

1. **Linear Algebra and Dot Products**

   We shall say that a linear transformation is **rigid** if each elementary vector is mapped to a unit length vector and the image of any pair of distinct elementary vectors is orthogonal.

   (a) Give a matrix definition of when a matrix is a rigid transformation.

   (b) Show that dot products are preserved under rigid transformations.

2. **Linear Algebra**

   Suppose that you are given the vectors $P_1, \ldots, P_k \in \mathbb{R}^d$ that define the subspace

   $$W = \{\alpha_1 P_1 + \cdots + \alpha_k P_k \mid \alpha_i \in \mathbb{R}\}$$

   show how to write W as the solution to a set of linear constraints, i.e.

   $$W = \{x \in \mathbb{R}^d \mid Ax = 0\}$$

   Suppose you are given a subspace as a set of constraints $Ax = 0$ show how one could write it as a set of generators.

3. **Convex Closure**

   We said that a set $C \subset \mathbb{R}^d$ is **convex** if for any two points p and q in C the line segment $[p, q]$ is contained in C. Let $Q = \{P_1, \ldots, P_k\} \subset \mathbb{R}^d$.

   In class we defined to objects:

   Definition 0.1. *(a) The convex closure $CC(Q)$ to be the smallest convex set containing Q.*

   (b) $ConvexComb(Q) = \{\alpha_1 p_1 + \cdots + \alpha_k p_k \mid \alpha_1 + \cdots + \alpha_k = 1$ and $\alpha_i \geq 0\}$

   Show that $CC(Q) = ConvexComb(Q)$.

4. **Size of Priority Queue in Sweep Line Algorithm**

   In our analysis of the the sweep line algorithm we said that the size of the priority queue can be at most $|P| + |I|$ which is at most $O(n^2)$. Thus we concluded that the cost per operation was at most $O(\log n)$. Find a better upper bound or find an example match our bound.