

**Problem Set 3 Solutions**

*Due: Wednesday, September 27, 2017 at noon*

**Problem 3.1 [Right Isosceles Triangle Range Searching].**

Describe and analyze a static data structure for storing a set of  $n$  2D points subject to the following query operation:

Given an axis-aligned right isosceles triangle (bounded by a horizontal, vertical, and diagonal line), report the  $k$  points in the triangle in  $O(\log n + k)$  time.

Your data structure should occupy  $O(n \text{ polylog } n)$  space.

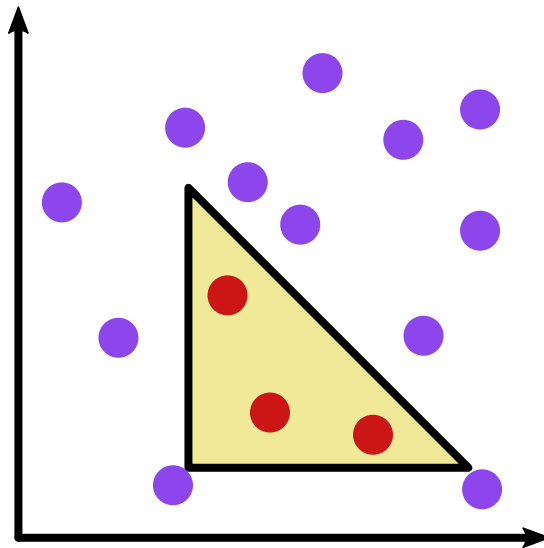


Figure 1: An example range query by an axis-aligned right isosceles triangle.

*Solution:* To solve this problem we will use a layered range tree with fractional cascading from Lecture 4 which supports orthogonal range search in three dimensions in  $O(n \lg n)$  time per query and  $O(n \lg^3 n)$  space for the data structure. For each point in our data set we compute a third ‘coordinate’  $z = x + y$ . Thus checking if  $z < c$  is the same as querying the inequality  $x + y < c$  which is precisely a 45 degree line. These can be calculated in  $O(n)$  time.

If the vertices on our triangle are  $\{(a_1, b_1), (a_2, b_1), (a_1, b_2)\}$  then we can make a query on our data structure with input  $([a_1, a_2], [b_1, b_2], [a_1 + b_1, a_2 + b_2])$ . This returns the points inside our desired triangle.