6.851 Advanced Data Structures (Spring'14)

Prof. Erik Demaine TAs: Timothy Kaler, Aaron Sidford

Problem 5 *Sample solution*

Due to an ambiguity in the problem statement we accepted solutions to two different problems: 1. Count the number of rectangles whose areas intersect, and 2. Count the number of rectangles whose boundaries intersect. Most solutions solved problem 1 whose solution is sketched below.

Counting all pairs of rectangles whose areas intersect can be reduced to solving two other counting problems: batched orthogonal range counting and axis-parallel line intersection counting.

This reduction comes from the observation is that for any two intersecting rectangles the sum of the corner-in-area intersections and the sum of rectangle-boundary intersections is exactly 4. Counting all corner-in-area intersections can be done using cache-oblivious batched orthogonal range queries using the rectangle corners as points and the rectangles as the set of range queries. Counting rectangle-boundary intersections can be done cache-obliviously by treating each rectangle as four axis-parallel lines and running the distribution sweeping algorithm described in [1].

The desired time bound in the cache-oblivious model follows from the analysis of lazy funnel sort.

References

[1] G. S. Brodal and R. Fagerberg. Cache oblivious distribution sweeping. Technical Report BRICS-RS-02-18, BRICS, Department of Computer Science, Aarhus University, 2009.