6.851 Advanced Data Structures (Spring'10)

Prof. Erik Demaine Dr. André Schulz TA: Aleksandar Zlateski

Problem 1 *Due: Thursday, Feb. 11*

Be sure to read the instructions on the assignments section of the class web page.

Transposing a matrix. Prove that there is a binary search tree supporting the access sequence $0, k, 2k, \ldots, k^2-k, 1, k+1, 2k+1, \ldots, k^2-k+1, \ldots\ldots, k-1, 2k-1, 3k-1, \ldots, k^2-1$ in constant amortized time per operation. (This access sequence is equivalent to accessing a $k \times k$ matrix of entries, in column-major order, when the items are stored in row-major order.)

Logarithmic redux. Give a purely geometric argument that there is a binary search tree supporting any m searches on n items in $O(m \log n)$ total time. (Reason about point sets, not binary search trees.)