This lecture covers our second of two lower bounds. This one is work by Mihai Pătraşcu (former 6.851 TA) and myself. We'll show that maintaining a graph subject to edge insertion, edge deletion, and connectivity queries (are \( v \) & \( w \) connected by a path?) requires \( \Omega(lg n) \) time per operation, even if the graph is just a bunch of paths. This in particular proves optimality of the logarithmic dynamic tree data structures, and shows that the \( O(lg^2 n) \) data structure we saw for general graphs is pretty good. The lower bound introduces a new but very simple technique, which at the time was the first “truly logarithmic” lower bound for a natural problem, yet the whole proof is relatively clean.
6.851 Advanced Data Structures
Spring 2012

For information about citing these materials or our Terms of Use, visit: http://ocw.mit.edu/terms.