

This lecture is the first of two about succinct data structures—data structures using very close to the minimum amount of space (“just the data”). In particular, we'll see how to store an n -node binary trie in $2n + o(n)$ bits (which is optimal up to the little-oh term) and still be able to navigate to a node's left child / right child / parent in constant time per operation, as well as be able to compute the size of the subtree rooted at a node in constant time. Along the way, we'll see how to store an n -bit string in $n + o(n)$ bits (which is again optimal up to little-oh) and be able to compute the number of 1 bits left of a given bit, and find the i th 1 bit, in constant time per operation. These data structures form the basis of many succinct data structures.

MIT OpenCourseWare
<http://ocw.mit.edu>

6.851 Advanced Data Structures
Spring 2012

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