This lecture marks our full entry into integer data structures (though hashing was also one), as well as our first of three lectures on the predecessor problem: supporting insert, delete, and predecessor/successor on a set of n w-bit integers. I'll give an overview of what's known about this problem, as well as the appropriate models of computation, and then proceed to the first main result: data structures achieving O(lg w) time per operation. When w is polylogarithmic in n (a common case), this bound is O(lg lg n), an exponential improvement over binary search trees. The first such data structure is called (and by) van Emde Boas, and it uses Θ(2w) space. A little addition of hashing reduces the space to optimal Θ(n). We'll also see simpler ways to achieve the same bounds, using a simple tree view, indirection, and structures called x-fast and y-fast trees by Willard (who was actually the first to get Θ(n) space).

6.851 Advanced Data Structures
Spring 2012