Design and Analysis of Algorithms: Lecture 4

Ben Chaplin

Contents

1 Van Emde Boas Trees

1.1 Operations

We want to maintain n elements from the set $\{0, 1, \ldots, u-1\}$, and support the following operations:

- insert(V, x): insert x into V
- delete(V, x): delete x from V
- successor(V, x): return the smallest element in V which is larger than x

1.2 Problem

Balanced binary search trees support all three of the above operations in $O(\log n)$ time. The goal of **van Emde Boas trees** is to support operations in $O(\log \log n)$ time.

Let n and u be defined as they were above. If $u = n^{O(1)}$, then $\log \log u = O(\log \log n)$.

1.3 Recurrences

Recall the binary search recurrence:

$$T(k) = T\left(\frac{k}{2}\right) + O(1) \tag{1}$$
$$= O(\log k) \tag{2}$$

So we're seeking:

$$T(\log u) = T\left(\frac{\log u}{2}\right) + O(1) \tag{3}$$
$$= O(\log \log u) \tag{4}$$

Which can be written:

$$T'(u) = T'(\sqrt{u}) + O(1) \tag{5}$$
$$= O(\log \log u) \tag{6}$$