第五次书面作业 平摊分析

1. Making binary search dynamic

Binary search of a sorted array takes logarithmic search time, but the time to insert a new element is linear in the size of the array. We can improve the time for insertion by keeping several sorted arrays.

Specifically, suppose that we wish to support SEARCH and INSERT on a set of n elements. Let $k = \lceil \lg(n+1) \rceil$, and let the binary representation of n be $\langle n_{k-1}, n_{k-2}, \ldots, n_0 \rangle$. We have k sorted arrays $A_0, A_1, \ldots, A_{k-1}$, where for $i = 0, 1, \ldots, k-1$, the length of array A_i is 2^i. Each array is either full or empty, depending on whether $n_i = 1$ or $n_i = 0$, respectively. The total number of elements held in all k arrays is therefore $\sum_{i=0}^{k-1} n_i \, 2^i = n$. Although each individual array is sorted, elements in different arrays bear no particular relationship to each other.

a. Describe how to perform the SEARCH operation for this data structure. Analyze its worst-case running time.

b. Describe how to perform the INSERT operation. Analyze its worst-case and amortized running times.

c. Discuss how to implement DELETE.

2. Move-to-front List

1) 我们猜想：如果已知访问序列，则在队列上，每次将当期访问的第 i 个元素移动到从当期时刻算起，在该元素下一次被访问前，所有被访问元素的后面，则可以获得最佳的访问开销。我们也不知道这个贪心的策略正确与否，请思考如何证明该贪心策略，或找出反例。

2) 如果考虑每次移动元素时，总能已常数时间开销将被访问的第 i 个元素移动到队列头。能否用势函数证明，此时其时间开销不大于最优策略时间开销的 2 倍？