习题2-11

TC-6.3.3. 6.4.2. 6.5.7.

6.3-3

Show that there are at most $\lceil n/2^{h+1} \rceil$ nodes of height h in any n-element heap.

- 数学归纳:
  - h=0时,显然成立
  - 假设h=k成立,即k层上至多$\lceil n/2^{k+1} \rceil$个节点
  - h=k+1时,k+1层上的节点数至多为k层的一半(k+1层所有节点均有孩子), 此时k+1层上的节点数至多为$\lceil \lceil n/2^{k+1} \rceil /2 \rceil = \lceil n/2^{k+2} \rceil$
  - 由数学归纳,原命题成立

6.4-2

Argue the correctness of HEAPSORT using the following loop invariant:

> At the start of each iteration of the **for** loop of lines 2–5, the subarray $A[1 \,..\, i]$ is a max-heap containing the i smallest elements of $A[1 \,..\, n]$, and the subarray $A[i+1 \,..\, n]$ contains the $n-i$ largest elements of $A[1 \,..\, n]$, sorted.

HEAPSORT(A)

1 BUILD-MAX-HEAP(A)

2 初始化:$i = n$时,$A[1 \cdots n]$是一个最大堆

3 **for** $i = A.length$ **downto** 2

4 循环不变式:在循环开始时,$A[1 \cdots i]$是一个包含了数组$A[1 \cdots n]$中i个小元素的最大堆,$A[i+1 \cdots n]$包含了数组$A[1 \cdots n]$中已排序的$n-i$个最大元素.

5   **do** exchange $A[1]$ whth $A[i]$

6         $A.heapsize = A.heapsize - 1$

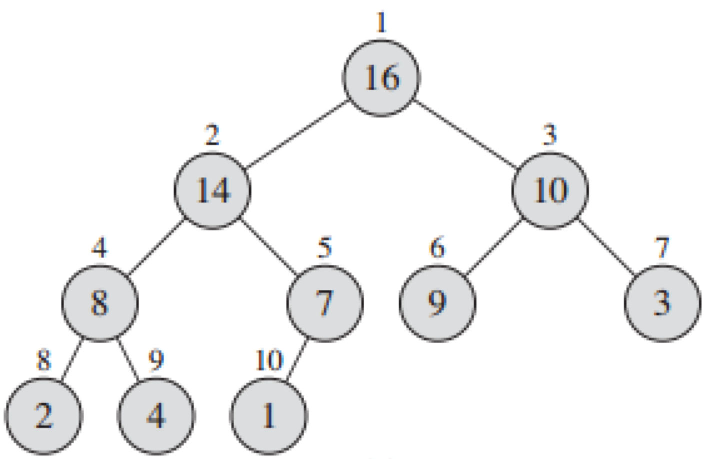
7   MAX-HEAPIFY$(A, 1)$

8 保持:把当前元素$(A[1 \cdots i])$中的最大值从堆中取出,与第i个位置交换,这样$A[i+1 \cdots n]$包含了数组$A[1 \cdots n]$中已排序的$n-(i-1)$个最大元素,再将前面$i-1$个小元素的最大堆维护一遍,这使循环的性质得到保持.

9 终止:此时最大堆中只有一个元素,且是n个元素的最小值,后$n-1$个元素以排好序,所以n个元素被排好序.

6.5-7

Show how to implement a first-in, first-out queue with a priority queue. Show how to implement a stack with a priority queue. (Queues and stacks are defined in Section 10.1.)



以元素添加时序
作为排序关键字