6-2 Analysis of d-ary heaps

A d-ary heap is like a binary heap , but (with one possible exception) non-leaf nodes have d children instead of 2 children.

a.How would you represent a d-ary heap in an array?

Ans:

假設 root 放在為 array[1]，他的 d 個 children 就存在 array[2]~array[d+1]，root 的第一個 child 的 d 個 children 就放在 array[d+2]~array[2d-1]依此類推。

對第 i 個 node 來說:

其 parent 為 $\left\lfloor \frac{i-2}{d} + 1 \right\rfloor$

其第 j 個 child 為 d(i-1)+j+1

b.What is the height of a d-ary heap of n elements in terms of n and d?

Ans:

$\Theta(\log_d n)$

c.Give an efficient implementation of EXTRACT-MAX in a d-ary max-heap . Analyze its running time in trems of d and n.

Ans:

類似 binary heap 的作法，return root，heap size 減一，在從最後面拿 node 放在 root，再與其 d 個 children 比較去做調整(作 heapify)。

running time=$\Theta(d * \log_d n)$

d.Give an efficient implementation of INSERT in a d-ary max-heap . Analyze its running time in terms of d and n.

Ans:

與 binary heap 類似，將 inset 的 node 放在 heap 最後面，heap size 加一，再與 node 的 parent 比較去做調整。

running time=$\Theta(\log_d n)$

e.Give an efficient implementation of INCREASE-KEY(A,i,k),which first sets A[i]←max(A[i],k) and then updates the d-ary max-heap structure appropriately. Analyze its running time in terms of d and n.

Ans:

如果 k<A[i]，flag an error

否則 set A[i]=k，再與其 parent 比較做調整。

running time=$\Theta(\log_d n)$