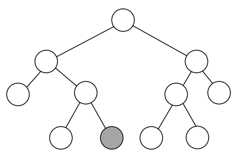
10 Algorithms 2 (djw1005)

Consider a Dictionary whose keys belong to a totally ordered set, and whose values are real numbers. We would like to implement an additional operation: `partialsum(`k`,`k'`)` which sums all values whose key ℓ satisfies $k \le \ell < k'$.

We can implement this Dictionary using a balanced binary search tree, and implement `partialsum` by first searching for k then calling `successor` until we reach a key $\ell \ge k'$ or we run out of keys. (The `successor` function, when applied to a node in the tree whose key is k, returns the node with the smallest key that is $> k$, if one exists.)

We can analyse the cost of `partialsum` by treating it as a sequence of operations: one search, then one or more calls to `successor`. We can analyse the cost of this sequence using the potential method.



(*a*) In the tree shown above, label nodes by the order in which they are visited by successive calls to `successor`, starting from the shaded node. [2 marks]

(*b*) Give pseudocode for the `successor` function. Show that the worst-case cost of `successor` is $\Omega(\log n)$, where n is the number of items in the tree. [5 marks]

(*c*) Consider the function

$$\Phi(k) = 2r_k + D - d_k$$

where D is the depth of the tree, r_k is the number of right-child steps on a path from root to the node with key k, and d_k is depth of that node. Augment Φ by defining its value at an 'initial empty' state, which you should define. Explain why your augmented function is a potential function. [3 marks]

(*d*) Show that `partialsum` is $O(m + \log n)$, where n is the number of items in the tree and m is the number of calls to `successor`. Explain your reasoning. [10 marks]