CPS 230 Midterm

Fall 1998

10:55am-12.10pm, Tuesday October 20th
Closed book exam

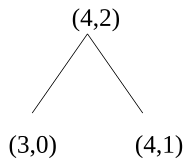
20 points Solve the recurrence equation:

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ \sqrt{n}T(\sqrt{n}) + n \lg n & \text{if } n \geq 2 \end{cases}$$

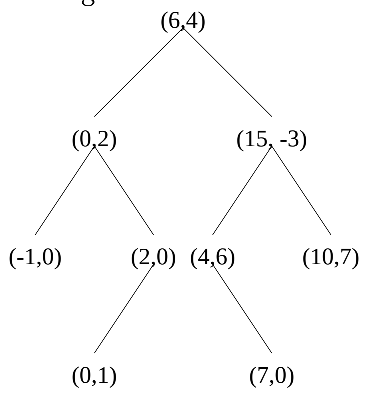
20 points Let S be an array of n (not necessarily distinct) integers. Describe an $O(n)$-time algorithm to test whether any item occurs more than $\lceil n/2 \rceil$ times in S.

30 points Consider a binary search tree in which each node v contains a key as well as an additional value called 'addend'. The addend of node v is implicitely added to all keys in the subtree rooted at v (including v). Let (key, addend) denote the contents of any node v.

For example, the following tree contains the elements {5,6,7}:



A) Which elements does the following tree contain?



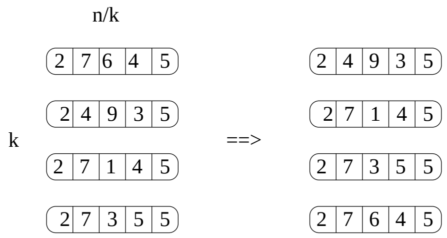
B) Let h be the height of a tree as defined above. Describe how to perform the following operations in $O(h)$ time:

- FIND(x, T): return YES if element x is stored in tree T
- INSERT(x, T): inserts element x in tree T
- PUSH(x, k, T): add k to all elements $\geq x$

C. Describe how it can be insured that $h = O(\log n)$ during the above operations. (Hint: show how to perform rotation)

30 points In this problem we consider the problem of sorting k strings of n/k integers each (thus overall the strings contain n integers). A string $X =< x_1, x_2, x_3, ..., x_{n/k} >$ is smaller than a string $Y =< y_1, y_2, y_3, ..., y_{n/k} >$ if $x_1 = y_1, x_2 = y_2, \ldots, x_{i-1} = y_{i-1}$ and $x_i < y_i$ for some $1 \leq i \leq n/k$.

Below is an example for k=4 strings with n/k=5 integers each.



Assume that to find the order of the strings we can only use comparisons between the individual integers of the strings (the *comparison model*).

A) Argue that we need $\Omega(k \log k + n)$ comparisons to sort the strings.

B) Consider the following sorting algorithm:

(a) Find the median m of the first integer in each of the strings using $O(k)$ comparisons (using the linear selection algorithm)

(b) Divide the strings into three groups using $O(k)$ comparisons: L consisting of all strings whose first integer is less than m; G consisting of all strings whose first integer is greater than m; E consisting of all strings whose first integer is equal to m

(c) Recursively sort L

(d) Recursively sort E, without considering the first integer in the strings.

(e) Recursively sort G

(f) Return the sorted set of strings consisting of the strings in L, followed by the strings in E, followed by the strings in G(Note that no comparisons are used in this step)

Analyse the worst-case number of comparisons used by the above algorithm. (Hint: Do not try to formulate and solve a recurrence relation)