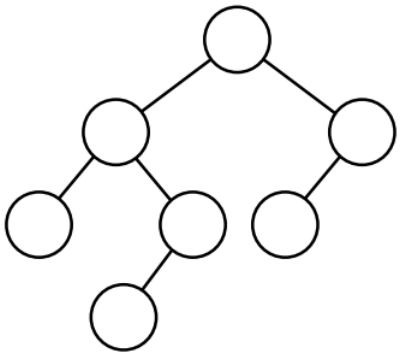
COS 226 Data Structures and Algorithms
Computer Science Department
Princeton University
Spring 2016

Week 5 Activity

1. BST's and LLRB's

   (a) Label each node in the following binary tree with numbers from the set $2, 26, 10, 27, 20, 15, 42$ so that it is a legal Binary Search Tree. (Hint: use some scratch space, and only write down the answer once you have it.)



   (b) Now label each edge in the figure with r or b, denoting RED and BLACK, so that the tree is a legal LLRB.

   (c) Argue that it is not possible to assign different red/black labels and still satisfy the LLRB-tree conditions.

2. Hashing short questions Answer each of the following questions by writing true, false or by choosing from multiple choice answer(s).

   (a) If hash table size is greater than or equal to the number of keys, then linear probing always finds a spot for any key.

   (b) Choosing a bad hash function can result in (choose all that apply)

      i. too many collisions

      ii. slow performance

      iii. no constant runtime guarantees

      iv. too much clustering when using linear probing

   (c) If a linear probing hash table is 50% full, the average number of probes required for a search hit is 1.5.

   (d) Searching in a hash table is always faster than searching in a LLRB.

   (e) It is possible to find the key with the maximum value in a hash table in worst case $\log_2 N$ time or less.

   (f) Which of the following statement(s) is/are true about linear probing hash tables (choose all that apply).

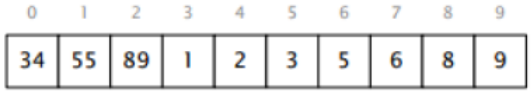
      i. deleting many keys can result in degraded performance

      ii. has better cache performance

      iii. easy to implement

   (g) Suppose that 10,000 random ASCII strings of length 5 is hashed into separate chaining hash table of size N=10,000 using the hash function $hashcode(s) = (s[0] + s[1] + s[2] + s[3] + s[4])N$ where $s[i]$ is the character i of the string s. What is the average length of a chain in the separate chaining hash table?

3. Algorithm Design Question (Bonus)
An array b is called a Circular Shift of array a, if b is obtained by rotating a sorted array a clockwise as shown below.



(a) Assume that the array b consists of N comparable keys, no two of which are equal. Array a is not provided. Design an efficient algorithm to determine the minimum value of array b. Briefly describe your algorithm, using crisp and concise prose.

(b) Design an efficient algorithm to find any given key in array b. You can use your algorithm in part (a) to help solve this problem. Briefly describe your algorithm, using crisp and concise prose.

4. Midterm Preparation

The following topics will be covered in the midterm exam. Be sure to review lecture notes, assignments, and blackboard exercises. The exam page is available at http://www.cs.princeton.edu/courses/archive/spring16/cos226/exams/midterm-info-fall.html You can find information about the exam location, review sessions and office hours before the midterm.

(a) *Union-find:* quick-find,quick-union,weighted quick-union

(b) *Elementary data structures:* resizing arrays,linked lists,stacks,queues

(c) *Elementary sorting algorithms:* insertion sort,selection sort,Knuth shuffle

(d) *Linearithmic sorting algorithms:* mergesort,bottom-up mergesort,quicksort,3-way quicksort,quickselect

(e) *Prioroty queues:* binary heaps,heapsort

(f) *Binary search and BSTs:* sequential search,binary search,BSTs

(g) *Balanced trees:* 2-3 trees,left-leaning red-black BSTs

(h) *Hashing:* separate chaining,linear probing