

Lab 9

Due Tuesday 9/17, 2 PM

1. Starting with an initially empty BST, imagine loading the BST using the following insertion sequence

$$a_1, a_2, \dots, a_s, b_1, b_2, \dots, b_t$$

where $a_1 > a_2 > \dots > a_s$ and $b_1 < b_2 < \dots < b_t$, and $a_1 < b_1$. For this BST, what is the worst-case asymptotic running time for searches and insertions?

2. Use the insertion algorithm for red-black trees to successively insert the values 3, 2, 1, 4, 5, 6 starting with an empty tree.
3. Carry out the array-based version of HeapSort on the input array [1, 4, 3, 9, 12, 2, 4]. Indicate clearly your steps and separate Phase I (heapification) from Phase II (in-place sorting).
4. *Interview Question.* Devise an algorithm to solve the following problem. Input is a set $S = \{s_0, s_1, \dots, s_{n-1}\}$ of positive integers and a non-negative integer k . Output is *true* if there is a subset T of S whose sum is k ; *false*, otherwise. What is the running time of your algorithm? Write your algorithm in pseudo-code.