Algorithms
Exam2 Review Questions

1. T F The height of any binary search tree with n nodes is O(log n).

2. How does the key in a node compare to the keys of its children in a binary search tree?

3. Consider using a hash table for integer values and resolving collisions by open addressing.

   At attempt i, you hash key k to the slot f(k) + i *g(k) (mod 11), where f(x) = x (mod 11) and g(x) = x^2 + 1 (mod 11). (The first attempt is attempt 0.) The table is initially empty (every slot has value NIL), and we perform the following operations in order:

   - Insert 3
   - Insert 14
   - Insert 90
   - Insert 2
   - Delete 14

   i. What are the contents of the table after the above sequence operations? You should fill in the contents in EVERY slot in the hash table with either DELETED, NIL, or an INTEGER VALUE.

   ii. At this point, if we search for 13, what is the sequence of slots that we check before returning that it does not exist in the table?

4. The following problems have you fill in the code for binary search tree algorithms. It is assumed that each node of a tree has a value, val, and pointers to its left, right, and parent nodes, which may be None. All node values are distinct.

   (a) Fill in the missing lines in the following code for next, which returns the next (successor) node in the binary search tree containing node, or None if there is none:

```
def next(node):
  if node.right is not None:
    cur_node = node.right
    while cur_node.left is not None:
      cur_node = cur_node.left
    return cur_node
  else:
    prev_node = node
    cur_node = node.parent
    while cur_node is not None and prev_node == _____:
      prev_node = cur_node
      cur_node = prev_node.parent
    return cur_node
```

   (b) If the tree containing node is balanced and contains n nodes, what is the asymptotic worst case runtime of this algorithm?

5. What is the running time of RADIX-SORT on an array of n integers in the range 0, 1 … n^5 -1 when using base-10 representation? What is the running time when using a base-n representation?

6. T F If you know the numbers stored in a BST and you know the structure of the tree, you can determine the value stored in each node.

7. T F When you double the size of a hash table, you can keep using the same hash function.

8. T F In a BST, we can find the next smallest element to a given element in O(1) time.

** You should also study Quiz#3 that covered “Sorting in linear time”.
** Homework 3 that covered “Hashing Table”

GOOD LUCK !!!!