EECS560 **Practice HW#2** **9/9/18**

Instruction: You must show all your work clearly for credit. Partial credit will only be given to meaningful answers.

1. Use the definition of big-O to prove that $n^4 - 8n^3 + 16n^2 - 3n + 560 \neq O(n^3)$.

2. For any two functions $f(n)$ and $g(n)$, by definition, $f(n) = o(g(n))$ iff $f(n) = O(g(n))$ and $f(n) \neq \Theta(g(n))$. Use the definition of little-o to prove or disprove that $n^4 - 8n^3 + 16n^2 - 3n + 560 = o(n^5)$.

3. Let $f(n)$ and $g(n)$ be any two positive functions. Prove or disprove the statement that if $f(n) = O(g(n))$, then $2^{f(n)} = O(2^{g(n)})$.

4. By assuming that all basic operations require the same constant cost *K*, compute the cost of the resource function $R_w(n)$ in closed-form for the following program segment using the simplified approach as discussed in class. *Remark:* You must first set up a summation equation for $R_w(n)$ and then evaluate the sum(s) clearly for credit.

```
x = 210;
y = 560;
for i = 1 to n*n do
  for  j = i to n do
       y = x * y / 660 + 388;
  endfor;
endfor;
```

5. Given a set of records with 7 keys S = {35, 28, 43, 17, 39, 3, 46}.
   (a) By using the hash function *h(x) = x mod m* and chaining with singly linked list in constructing an open hash table H with m = 11 buckets, insert the records in S, in the given order, into H. You must show your computations for locations and illustrate the final structure of your hash table H clearly for credit. Remark: Insertion must be done at the beginning of the list.
   (b) By using the hash function *h(x) = x mod m* and quadratic probing in constructing a closed hash table H with m = 11 buckets, insert the records in S, in the given order, into H. You must show your computations for locations and illustrate the final structure of your hash table H clearly for credit.
   (c) Given two hash functions *h(x) = x mod m* and $h^+(x) = p - x \bmod p$. By using open addressing with $f_i = i * h^+(x)$ and double hashing in constructing a closed hash table H with m = 11 buckets and p = 5, insert the records in S, in the given order, into H. You must show your computations for locations and illustrate the final structure of your hash table H clearly for credit.

6. If a set of 4090 records is being stored using a binary tree T with 4090 nodes (one record per node), answer the following questions with ***integer solution*** if possible.
   (a) What is the min height of T?
   (b) What is the max height of T?
   (c) What is the min number of leaves in T?
   (d) What is the max number of leaves in T?
   (e) If T is being implemented using the sequential array data structure, what is the size of an array A in order to store T?

7. Construct the (unique) binary tree corresponding to the given pair of tree traversals if possible. **Remark:** You must show all your steps clearly as illustrated in class for credit. If no such tree is possible, you must justify your answer.

(a)	Preorder:	G B D F A I H J K L E C
	Inorder:	B I A F K J H L D E G C
(b)	Postorder:	H I B C A K G E J D F
	Inorder:	I H C B K G J E F D A
(c)	Postorder:	H G F B K L J I D A
	Inorder:	F G H B A D K J L I

8. Given a set S of 4 records with keys $\{x_1, x_2, x_3, x_4\}$, $x_1 < x_2 < x_3 < x_4$. Construct all possible binary search trees (BST) that can be used to store S. Remark: You must illustrate all your BSTs clearly for credit.

9. Given a set of 10 records with priorities S = {10, 15, 3, 8, 20, 5, 17, 27, 19, 12}.
   (a) Construct a BST T for S by inserting the records, in the given order, into an initially empty binary search tree. When done, delete 10, and then 20 from the tree.
   (b) Construct a BST T for S by inserting the records, in the reverse given order, into an initially empty binary search tree. When done, delete 5, and then 19 from the tree.

   Remark: You must show your BST after each insertion/deletion for credits. For deletion, you must use deleteMin operations as discussed in class.