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 $n^4 - 8n^3 + 16n^2 - 3n + 560 \neq O(n^3)$.

   Proof: If true, we must have constants k and n_0 such that $n^4 - 8n^3 + 16n^2 - 3n + 560 \leq n^3$, $\forall n \geq n_0$. Hence, $\dfrac{n^4 - 8n^3 + 16n^2 - 3n + 560}{n^3} \leq k$. As $n \to \infty$, we have $\infty \leq k =$ constant. A contradiction is reached, implying 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 $n^4 - 8n^3 + 16n^2 - 3n + 560 = o(n^5)$.

   Proof: (i) Prove that $n^4 - 8n^3 + 16n^2 - 3n + 560 = O(n^5)$.

   $n^4 - 8n^3 + 16n^2 - 3n + 560$

   $\leq n^4 + 16n^4 + 560n^4, \forall n \geq 1$

   $\leq 577n^4, \forall n \geq 1$

   $\leq 577n^5, \forall n \geq 1.$

   Hence, $n^4 - 8n^3 + 16n^2 - 3n + 560 = O(n^5)$.

   (ii) Prove that $n^4 - 8n^3 + 16n^2 - 3n + 560 \neq \Theta(n^5)$.

   If true, we must have $n^4 - 8n^3 + 16n^2 - 3n + 560 = \Omega(n^5)$, implying that there exist constants k and n_0 such that $n^4 - 8n^3 + 16n^2 - 3n + 560 \geq kn^5, \forall n \geq n_0$.

   Hence, $\dfrac{n^4 - 8n^3 + 16n^2 - 3n + 560}{n^5} \geq k, \forall n \geq n_0$.

   As $n \to \infty$, we have $0 \geq k > 0$, which is a contradiction.

   Hence, $n^4 - 8n^3 + 16n^2 - 3n + 560 \neq \Theta(n^5)$.

   Conclusion: $n^4 - 8n^3 + 16n^2 - 3n + 560 = o(n^5)$.

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 $2^{f(n)} = O(2^{g(n)})$.

   Solution:

   If $f(n) = O(g(n))$, then $2^{f(n)} = O(2^{g(n)})$ may or may not be true.

   Take $f(n) = 2n, g(n) = n$.

   $2^{2n} = 4^n \neq O(2^n)$.

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;
```

Wrong Solution:

$$T(n) = \sum_{i=1}^{n^2} \sum_{j=i}^{n} K$$

$$= K \sum_{i=1}^{n^2} (n - i + 1)$$

$$= K[n^3 - \frac{n^2(n^2+1)}{2} + n^2]$$

< 0. (Q: What is wrong?)

Correct Solution:

$$T(n) = ((\sum_{i=1}^{n} + \sum_{i=n+1}^{n^2}) \sum_{j=i}^{n}) K$$

$$= \sum_{i=1}^{n} \sum_{j=i}^{n} K + \sum_{i=n+1}^{n^2} K$$

$$= K \sum_{i=1}^{n} (n - i + 1) + (n^2 - n) K$$

$$= K [2n^2 - \frac{n(n+1)}{2}]$$

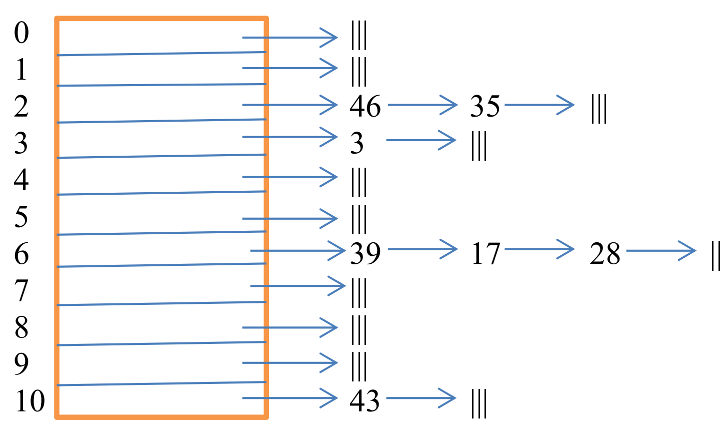
$$= K (\frac{3n^2 - n}{2})$$

$$= \Theta(n^2).$$

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.

   Solution:

   (a) Address computations:
   35 % 11 = 2,
   28 % 11 = 6,
   43 % 11 = 10,
   17 % 11 = 6,
   39 % 11 = 6,
   3 % 11 = 3,
   46 % 11 = 2.



Open hash table with chaining

(b) Address computations:

$35 \% 11 = 2$,
$28 \% 11 = 6$,
$43 \% 11 = 10$,
$17 \% 11 = 6 \rightarrow 7$,
$39 \% 11 = 6 \rightarrow 7 \rightarrow 10 \rightarrow 15 \% 11 = 4$,
$3 \% 11 = 3$,
$46 \% 11 = 2 \rightarrow 3 \rightarrow 6 \rightarrow 11 \% 11 = 0$.

0	**46**
1	
2	**35**
3	**3**
4	**39**
5	
6	**28**
7	**17**
8	
9	
10	**43**

Closed hash table with quadratic probing

(c) Address computations:

$35 \% 11 = 2$,
$28 \% 11 = 6$,
$43 \% 11 = 10$,

$17 \% 11 = 6$,
$h^+(x) = p - (x \bmod p) = 5 - (17 \bmod 5) = 3$,
$h_1(x) = (h(x) + 1h^+(x)) \bmod 11 = (6 + 3) \bmod 11 = 9$.

$39 \% 11 = 6$,
$h^+(x) = p - (x \bmod p) = 5 - (39 \bmod 5) = 1$,
$h_1(x) = (h(x) + 1h^+(x)) \bmod 11 = (6 + 1) \bmod 11 = 7$.

$3 \% 11 = 3$,

$46 \% 11 = 2$,
$h^+(x) = p - (x \bmod p) = 5 - (46 \bmod 5) = 4$,
$h_1(x) = (h(x) + 1h^+(x)) \bmod 11 = (2 + 4) \bmod 11 = 6$,
$h_2(x) = (h(x) + 2h^+(x)) \bmod 11 = (2 + 8) \bmod 11 = 1$.

0	
1	**46**
2	**35**
3	**3**
4	
5	
6	**28**
7	**39**
8	
9	**17**
10	**43**

Closed hash table with double hashing

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?

Solution:

$4090 = 2^{12} - 6$.

(a) What is the min height of T?

$\lfloor \mathbf{lgn} \rfloor = \mathbf{11}$

(b) What is the max height of T?

$n - 1 = 4089$.

(C) What is the min number of leaf in T?

1 (skew tree)

(d) What is the max number of leaf in T?

Height h = 11 implies tree with $2^{11+1} - 1 = 4095$ nodes and $2^{11} = 2048$ leaves.
Remove 5 leaves at level h and add back 2 new leaves at level h-1:

2048 – 5 + 2 = 2045 leaves.

(e) If T is being implemented using the sequential array implementation, what is the size of array A in order to store T?

$\mathbf{2^{4089+1} - 1 = 2^{4090} - 1}$

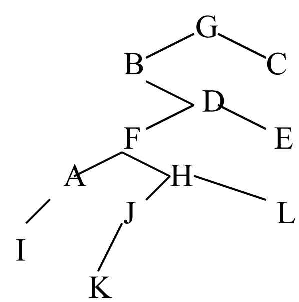
7. Construct the (unique) binary tree corresponding to the given pair of tree traversals if possible.
Remark: You must show all your steps clearly as illustrated in class for credit. If no such a tree is possible, you must justify your answer.
(a) Preorder: G B D F A I H J K L E C
Inorder: B I A F K 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

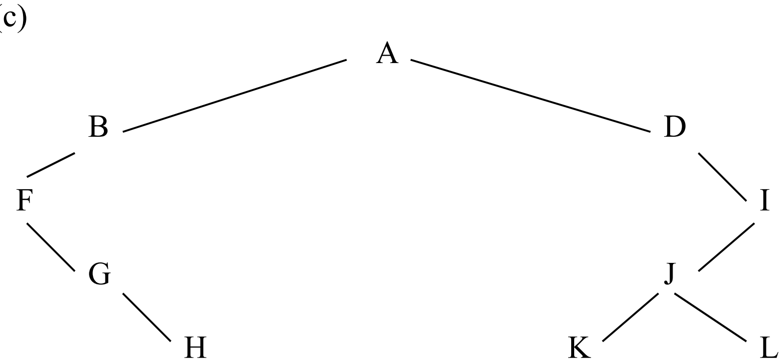
Solution:

(a)



(b) No tree.

(c)



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.

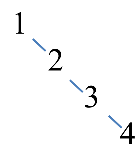
Solution:

Number of BST = $C_{n+1} = \frac{1}{n+1}\binom{2n}{n}$, which is the (n+1)th-term Catalan number.

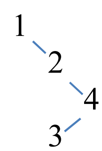
For n = 4, $C_5 = \frac{1}{5}\binom{8}{4} = 14$.

Distinct BST's for 4 records:

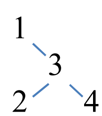
1:



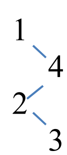
2:



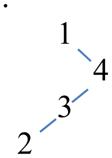
3:



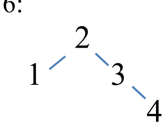
4:



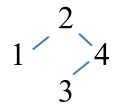
5:



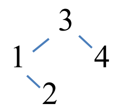
6:



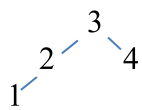
7:



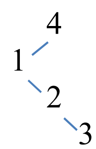
8:



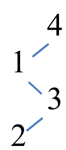
9:



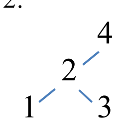
10:



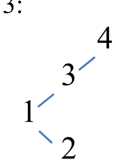
11:



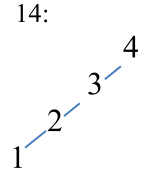
12:



13:



14:



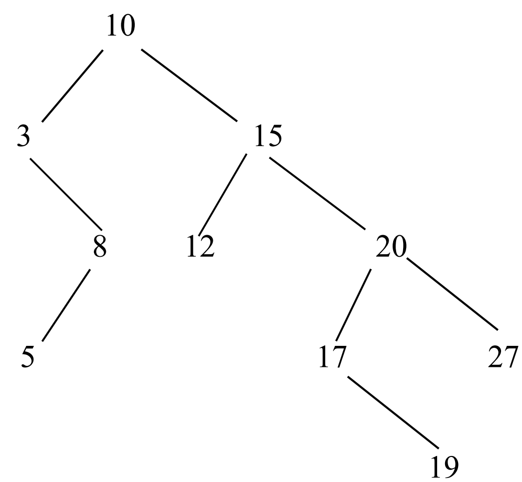
9. Given a set of 10 records with priorities S = {10, 15, 3, 8, 20, 5, 17, 27, 19, 12}.
   (a) Construct a BST T for S by inserting the records, in the given order, into an initially empty binary search tree. When done, delete 10, and then 20 from the tree.
   (b) Construct a BST T for S by inserting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

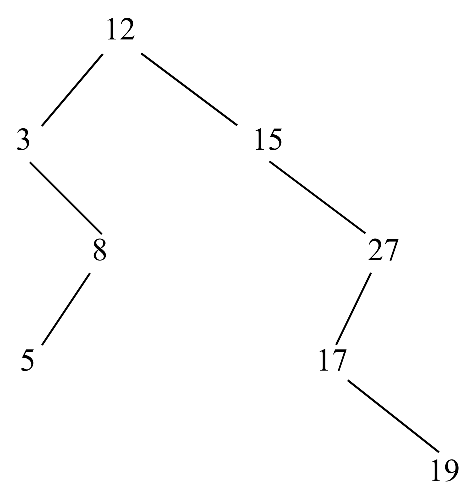
Solution:

(a)

After insertions:

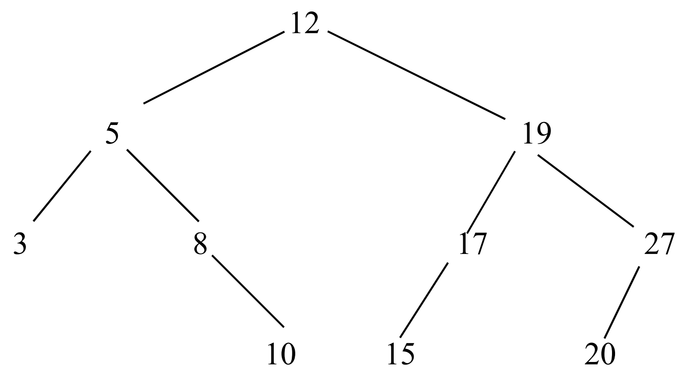


After deletions:



(b)

After insertions:



After deletions: