Instructor: Dr. Sartaj Sahni
Summer, 2003

Advanced Data Structures
(COP 5536 /AD 711R)
Exam 2

CLOSED BOOK
60 Minutes

Name: _____

NOTE:
1. **For all problems, use only the algorithms discussed in class/text.**
2. All answers will be graded on correctness, efficiency, clarity, elegance and other normal criteria that determine quality.
3. The points assigned to each question are provided in parentheses.

1. (12) For *min Fibonacci heaps*,

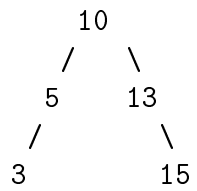
   (a) (3) Construct a min *Fibonacci heap* using *insert* operations with the following key sequence 2, 4, 6, 8, 10, 12, 14, 16, and 18. The *min Fibonacci heap* is initially empty.

   (b) (5) Perform a $RemoveMin$ operation on the *resulting Fibonacci heap* of (a), clearly label $ChildCut$ values.

   (c) (4) Perform a $DecreaseKey$ operation by changing 16 to 3 on the resulting *Fibonacci heap* of (b). Draw the resulting *Fibonacci heap*, clearly label $ChildCut$ values.

2. (15) For AVL trees,

   (a) (6) Consider the AVL tree below:

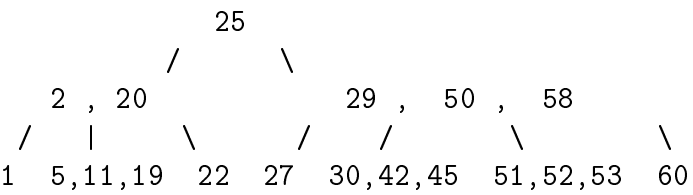


   Perform the following operations in this order. Show each step and specify the rotation type for each operation whenever a rotation is occurred.

$$insert(14),\ insert(1),\ remove(13),\ remove(15),\ remove(14)$$

   (b) (9) Recall that inserting a key into an AVL tree may require an RL rotation. Describe the three possible cases for an RL rotation by drawing, for each case, the tree before the insertion, just after the insertion, and after the rebalancing.

3. (10) For the following 2-3-4 tree,



(a) (5) Draw the *red-black* tree that results when the 2-3-4 to red-black transformation discussed in class/text are performed.

(b) (5) Draw the 2-3-4 tree that results from inserting 31 into the original 2-3-4 tree using $Two\text{-}Pass$ operations.

4. (13) For *red-black* trees, use the *bottom-up* algorithm for this problem.

   (a) (8) Construct a *red-black* tree by inserting the keys in the following sequence into an initially empty *red-black* tree : 1, 2, 3, 6, 4, 8, and 5.
   Show each step.

   (b) (5) For the resulting *red-black* tree of (a), perform a *Delete* operation for key value 3. Show each step and rotation type.