Instructor: Dr. Sartaj Sahni
Spring, 2002

Advanced Data Structures
(COP 5536 /AD 711R)
Exam 1

CLOSED BOOK
50 Minutes
Take one Week after Lecture 13

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. (10) Consider three queue operations, $insert(x, Q)$, $delete(Q)$, and $multiDelete(k, Q)$: $insert(x, Q)$ inserts the element x into the queue Q; $delete(Q)$ deletes and returns the first element of the queue Q; $multiDelete(k, Q)$ deletes and returns the first k elements of the queue Q (if there are less than k elements in Q, then it deletes and returns all the elements in the queue Q). An $insert(x, Q)$ and a $delete(Q)$ operation take $O(1)$ time each and a $multiDelete(k, Q)$ operation takes $O(k)$ time. The queue is initially empty.
   Show that the amortized complexity of each operation is $O(1)$.

2. (12) You are given 9 runs with 10, 20, 30, 40, 50, 60, 70, 80, and 90 equal-length records. It takes 10 seconds to read or write one block from/to disk and it takes 1 second to merge one block of records. Assume that all input, output, and CPU processing is sequential and there is no waiting time. The block size is 10 records.

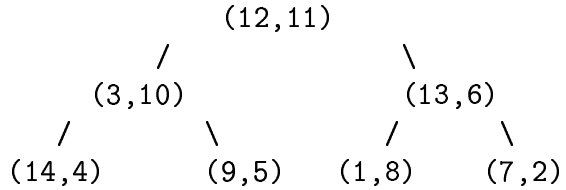
   (a) (4) Describe an optimal 4-way merging scheme.

   (b) (4) What is the total number of comparsions when a loser tree is used for merging ? (Justify your answer)

   (c) (4) Compute the total time taken by the optimal scheme.

3. *(8)*

(a) (5) Convert the following tree into an *interval heap*, showing each step.

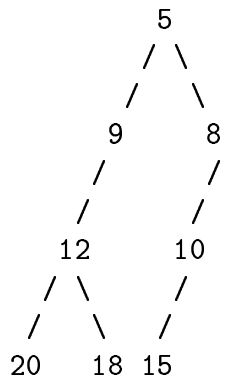


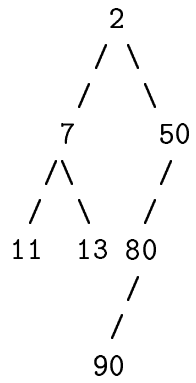
(b) (3) Perform $RemoveMin$ from the resulting interval heap of (a), showing each step.

4. *(10)* Consider a min leftist tree.

   (a) (4) *Convert* the following min trees to *min leftist trees* if necessary and label each node x with its $shortest(x)$ value.

   (b) (6) Draw the min leftist tree that results when the *meld* operation is performed on the two resulting min leftist trees. (*showing each step*)





5. *(10)* Show the *binomial heap* that results when the min node is deleted from the following binomial heap.(*showing each step*)
(Note: For consistency in solution, if you need to "combine" three binomial trees of the same size in the intermediate steps, please leave the binomial tree with the smallest root, and combine the other binomial trees.)