Instructor: Dr. Sartaj Sahni
Spring, 2003

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

CLOSED BOOK
60 Minutes
Take one Week after Lecture 13 (Feb. 12th 2003)

Name: ______________________________

SSN: ______________________________

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) Suppose that a sequence of n operations is performed on a data structure. The kth operation has a cost of $2\sqrt{k}$ whenever k is a perfect square (k = 1, 4, 9, 16, 25, etc), and otherwise it has a cost of 1. Use any one of the following methods of analysis to determine the amortized cost per operation: aggregate, accounting, or potential method. Please specify which method you are using.
Find the smallest integer amortized cost.

2. (8) You are given 5 runs with 500, 700, 900, 1100, 1500 equal-length records to be merged into one. 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 size of a block is 100 records.

   (a) (5) Describe an optimal 4-way merging scheme to merge the 5 runs into 1.

   (b) (3) Compute the total time taken by the optimal scheme of (a).

3. (10) For *interval heaps*,

(a) (5) Convert the following tree into an *interval heap*. Some intermediate steps to justify your answer must be provided.

```
                    (70,39)
                  /         \
          (99,82)            (1,12)
         /        \          /    \
     (20,23)      (48,20)  (49,63) (35,50)
     /     \       /
(25,19)(57,50) (46,19)
```

(b) (5) Perform a *removeMin* operation from the result interval heap of (a). Show the final interval heap as well as some of the intermediate steps.

4. (6) *Meld* the following height-biased min leftist trees, showing each step.

```
          3                   4
        /     \             /   \
      6         8          7     5
    /   \                 /
   9     11              10
```

5. (14) Start with an empty *min binomial heap*,

(a) (3) *Insert* the following sequence of keys into the *min binomial heap*:

7,12,4,22,3,24,13,23,19,5

Show the resulting structure.

(b) (5) Perform a *RemoveMin* operation on the heap of (a) (showing each step)
(Note: For consistency in solution, if you have three binomial trees of the same size in the intermediate steps, please leave the binomial tree with the largest root, and combine the other binomial trees.)

(c) (6) Prove that the *binomial* tree B_k (k is the degree of the tree) has the 2^k nodes ($k \geq 0$).