CSE 373 – Data Structures and Algorithms
Autumn 2003. Dry assignment #4
Due date: 11/14/03

1. (*20 pts.*) Given the following initial array of integers, show the contents of the array after each of the operations below.

$-\infty$	4	10	3	9	2	8	1	11	5	7

Note: While it is enough to show the content of the array after each operation, it is recommended to list some intermediate states too – this way you can still get partial credit if you make a mistake along the way. If you provide intermediate states, be sure to clearly mark the final state after each operation.

a. BuildHeap( );

b. DeleteMin( );

c. IncreaseKey (5, 7) [Here, 5 is the index of the element whose key should be increased by 7. Indices are counted from 0.]

d. Insert (1);

e. Insert (6);

2. (*20 pts.*) Recall that a maximum binary heap is one in which the key of each node is *larger* than the keys of its children. Given a complete tree T (like the one used to represent a binary heap), for each of the following claims determine if it is true or false. Prove or give a counter-example.

2a. If T represents a maximum binary heap, then a post-order traversal of T visits the nodes in an increasing key order: starting from the one with the minimal value and ending with the one with the maximal value.

2b. If a post-order traversal of T visits the nodes in an increasing key order – starting from the one with the minimal value and ending with the one with the maximal value – then T represents a maximum binary heap.

3. *(20 pts.)* Prove that a binomial tree B_k has binomial trees B_0, B_1, …, B_{k-1} as children of the root.

4. *(20 pts.)* Draw a binomial queue that includes the first 13 odd numbers 1, 3, 5, …, 25, and a binomial queue that includes the first 7 even numbers 2, 4, 6, …, 14. (There are many ways to do it – you need to show one way.) Then, merge the two binomial queues and draw the resulting binomial queue.

5. (*20 pts.*) A set includes the numbers 9, 21, 8, 25, 26, 12, 13. These numbers are inserted into a hash-table of size 13, using the hash function h(x)=x mod 13. Collisions are resolved using linear probing, that is, $h_i(x)=h(x)+i$. The insertion order of the numbers into the hash-table is not known (i.e., it is not known which number was inserted first, which one was inserted second, etc.). What is known is that after all the numbers have been inserted, the content of the table was:

index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
content	**13**	**26**	**25**						**8**	**21**	**9**		**12**

For each of the claims 5.1-5.4, determine if it:

(i) holds for any insertion order that produces the above table, or if

(ii) there is no insertion order that produces the above table for which the claim holds, or if

(iii) there exists an insertion order that produces the above table for which the claim holds and another insertion order that produces the above table for which the claim does not hold.

Justify your answers!

5.1 The number 25 was inserted last.

5.2 At least three numbers were inserted before 25.

5.3 8 was inserted before 12.

5.4 9 was inserted before 8.