

**CS483 - Midterm Exam**

Wednesday, February 28th

Prof. Jana Košecká

(100 points)

Please turn in your crib sheet with your exam.

Name: .....

Question	Score	Points
1	5	
2	5	
3	10	
4	25	
5	5	
6	10	
7	10	
8	15	
9	15	
Total	100	

1. (5) Using formal definition of  $\Omega(f(n))$  prove or disprove following relationship (choose the constants and show that they work):

$$n(n - 1) + 4 = \Omega(n^2)$$

2. (5) Provide an asymptotically tight bound ( $\Theta$ ) for the following recurrence:

$$T(n) = T(n - 2) + 1$$

3. (10) (a) Using Master's Theorem determine the asymptotic complexity of the following recurrence relation:

$$T(n) = 4T\left(\frac{n}{2}\right) + \lg n$$

- (b) Draw three levels of the recurrence tree corresponding to the above recurrence equation.

4. (25) Answer True or False. Justify your answer.

(10) If we know that in all input arrays, only adjacent elements can be out of order (i.e. if  $i \leq j$  and  $A[i] > A[j]$  then  $j = i + 1$ ), then it is better to use `InsertionSort` over `MergeSort` as the asymptotic performance of `InsertionSort` is better in this case. T F

(8) To build a heap of  $n$  elements takes  $\Omega(n \log n)$  time. T F

(7) INSERT operation can be made to run in  $O(\lg n)$  on a binary search tree with  $n$  elements. T F

5. (5) Briefly describe what do we mean by a randomized algorithm and name two examples.

6. (10) The following question pertains to BUILDHEAP procedure used in HEAPSORT.

```
procedure Buildheap(A)
```

```
1. for  $i := n/2$  downto 1 do
```

```
2.     Heapify(A,i)
```

a) Apply BUILDHEAP algorithm to the following array [3,1,6,12]. Draw the data structure corresponding to the initial array and then outline the steps of the BUILD-HEAP procedure.

b) What does HEAPIFY(A,i) accomplish ?

7. (10) Draw a binary search tree obtained by insertion of the elements in the following order 10, 7, 15, 12, 14, 20, 18.

Show the above binary search after deletion of element 15. Describe in words the steps of the deletion procedure.

8. (15) Given an array  $A$  of  $n$  real numbers in sorted order and another number  $x$ , how will you check in  $O(n)$  time if  $A$  has two elements whose sum is  $x$ ? Describe in words and outline the steps of your algorithm.

9. (15) Describe an algorithm that given  $n$  integers in the range of 1 to  $k$  processes its input and then answers a query about how many of the  $n$  integers fall into a range  $a..b$  in  $O(1)$  time. Your algorithm should use  $O(n + k)$  processing time.