UNIVERSITY OF MINNESOTA
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
4041H: HONORS ALGORITHMS AND DATA STRUCTURES
FALL 2015

MIDTERM EXAM (180 points):

This is a open book, open notes (including e-book and e-notes) exam. You may not use the internet or get assistance from others. You have 75 minutes to complete the exam.

Problem 1. (30 points)
Suppose you have the following priority min-heap:
[4, 9, 7, 10, 12, 8, 11, 15, 32]

(a). Decrease the key of "32" to "2" and recompile the min-heap in O(lg n) time. Show your work.

(b). After applying part (a), add a new element "3" to the heap in O(lg n) time. Show your work.

(c). After parts (a) and (b), remove the minimum element and recompile the min-heap in O(lg n) time. Show your work.

Problem 2. (20 points)
Suppose you had two (full binary tree) max-heaps, what is an efficient way to merge both? Give a concise description or write loose pseudocode **and give the running time**. If you use another algorithm as a sub-process, you need only explain what you use and not copy pseudocode from that algorithm.

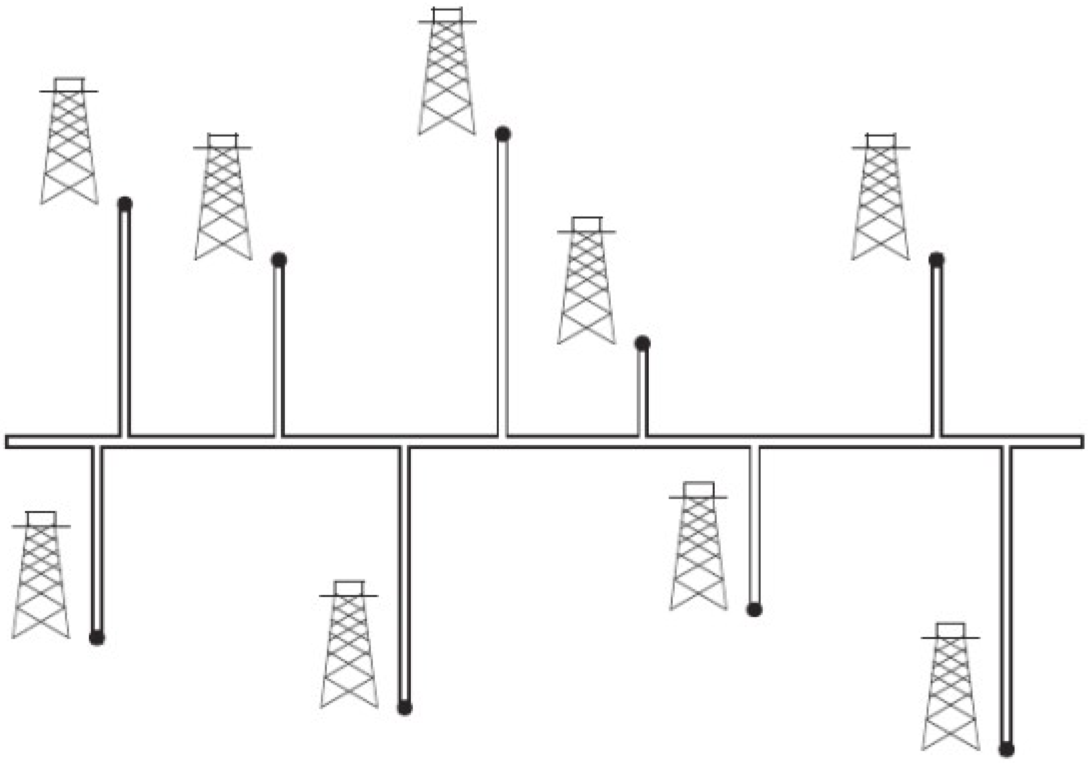
Problem 3. (20 points)
Use bucket-sort to lexicographically ("a" < "b" < "c" < "d") order the following strings (show work):
ddba, acba, abcd, dcac, ddcab, bdbab, aaaaa, cabab, daaca, bbdc, bbab, baaba

Problem 4. (30 points)
Given a sequence S, write some loose pseudocode to find if there is a majority element (over half of all elements in S is this element) in S. This algorithm must use o(n) space. If you use another algorithm as a sub-process, you need only explain what you use and do not need to copy pseudocode from that algorithm.

Problem 5. (30 points)
What is the running time of the following algorithm and explain why: (Hint: one way to start approaching this is to count how many times it compares "A[1] < A[2]?")

```
CrapSort(A):
for i = 1 to |A|
        for j = 2 to |A|
                if A[i] < A[j]
                        swap A[i] and A[j]
                        return CrapSort(A)
```

Problem 6. (30 points)
A oil company wants to build its main pipeline west to east and have multiple north/south pipelines connecting to each well. Given (x,y) coordinates for each well, **find an efficient algorithm** to decide where the main pipeline should be built.



Problem 7. (20 points)
Find the running time T(n) for each of the following algorithms that have a recursive relationship:

(a). $T(n) = 4\,T(n/2) + n^2 + n + 8$

(b). $T(n) = T(\,5n/7\,) + O(1)$

(c). $T(n) = 3\,T(n/2) + O(n^2)$

(d). $T(n) = T(\,n/10\,)$