University of New Mexico
Department of Computer Science

Final Examination

CS 561 Data Structures and Algorithms
Fall, 2007

Name:
Email:

- Print your name and email, *neatly* in the space provided above; print your name at the upper right corner of *every* page. Please print legibly.
- This is an *closed book* exam. You are permitted to use *only* two pages of "cheat sheets" that you have brought to the exam and a calculator. *Nothing else is permitted.*
- Do all the problems in this booklet. *Show your work!* You will not get partial credit if we cannot figure out how you arrived at your answer.
- Write your answers in the space provided for the corresponding problem. Let us know if you need more paper.
- Don't spend too much time on any single problem. The questions are weighted equally. If you get stuck, move on to something else and come back later.
- If any question is unclear, ask us for clarification.

Question	Points	Score	Grader
1	20		
2	20		
3	20		
4	20		
5	20		
Total	100		

1. **Short Answer**

For each problem below, give the answer in terms of simplest Θ. Please show your work where appropriate. (2 points each).

(a) Worst case time to find the minimum element in a min-heap

(b) Worst case time to find the maximum element in a min-heap

(c) Worst case cost of n calls to either Insert or Delete on the most efficient implementation of the dynamic table that we discussed in class

(d) $\sum_{i=1}^{n} \log i$

(e) Time to find a minimum spanning tree in a graph with n nodes and $\Theta(n^2)$ edges, using the fastest algorithm discussed in class

(f) Time to find if a cycle is reachable from some node v in a directed graph with n nodes and $O(n)$ edges.

(g) Solution to the recurrence $T(n) = 2T(n/2) + \sqrt{n}$

(h) Expected number of empty bins if you randomly throw $2n$ balls into n bins.

(i) What is the expected time to find the successor of some key (i.e. the node that comes right after the key) in a skiplist containing n items?

(j) Solution to the recurrence $T(n) = 2T(n-1) - T(n-2) + n$ (assume all constants in the solution are greater than 0)

2. **Short Answer (10 points each) Where appropriate, circle your final answer.**

   (a) Professor Plum postulates that if every edge on an undirected graph has a unique positive weight, then the shortest path tree rooted at v on that graph is always the same as the minimum spanning tree found by Prim's algorithm when seeded initially with the vertex v. Is this correct? If so, prove it. If not, give a counter example.

(b) Consider a data structure over an initially empty list that supports the following two operations. APPEND-NUMBER(x): Adds the number x to the beginning of the list; and MIN-MAX: Traverses the list, computing the minimum and maximum element in the list, and then creates a new list that contains only two numbers, the minimum and maximum of the old list.

- Assume an arbitrary sequence of n operations are performed on this data structure. What is the worst case run time of any particular operation?

- Show that the amortized cost of an operation is $O(1)$ using the potential method. Make sure to prove your potential function is valid.

3. **Segmentation**

In the segmentation problem, you want to segment text that is written without spaces into individual words. For example, if you are given the text "meetateight", the best segmentation is "meet at eight", not "me et at eight" or "meet ate ight". Assume you are given as a black box a function q that takes a string of letters x and returns $q(x)$, which gives a measure of how likely x is to be an English word. The quality of x can be positive or negative so that $q("me")$ is positive, and $q("ight")$ is negative. Given a string s, a segmentation of s is a partition of the letters into contiguous blocks. The total quality of a segmentation is the sum of the quality of each of the blocks of letters. So the quality of our segmentation above is $q("meet") + q("at") + q("eight")$. Give an efficient algorithm that takes as input a string s of length n and returns a segmentation of maximum total quality (assume that a single call to the q function takes constant time). Analyze your algorithm.

4. **Goods Trading**

Assume there are n goods $g_1, g_2, ..., g_n$ and there is an n by n table T such that one unit of good g_i buys $T[i, j]$ units of good g_j. Part 1: Give an efficient algorithm to determine whether or not there is a sequence of goods $g_{i1}, g_{i2}, ..., g_{ik}$ such that $T[g_{i1}, g_{i2}] * T[g_{i2}, g_{i3}] * ... * T[g_{ik}, g_{i1}] > 1$. In other words, give an algorithm to determine if there is a sequence of goods that can be traded to actually obtain more of some good. Analyze your algorithm.

Part 2: Give an efficient algorithm to print out such a sequence of goods if one exists. Analyze your algorithm.

5. **Bad Santa**

A Bad Santa has hidden $n/2$ Nintendo Wii's in n boxes. A child is presented with each box in sequence and must decide to either open that box immediately or to pass on the box and never be able to open it again. The child wants to guarantee she get at least one Wii while opening the smallest number of boxes in expectation. The Bad Santa knows the child's algorithm and places the Wii's in boxes so as to try to maximize the expected number of boxes opened. In this problem, you must design and analyze an algorithm for the child that 1) guarantees that the child finds a Wii; and 2) minimizes the expected boxes opened until that Wii is found.

Hint 1: Birthday paradox; Hint 2: You may again find the following inequality useful $1 - x \leq e^{-x}$

A final note, not necessary for solving the problem: this problem has applications in designing power-efficient sensor networks (the boxes are time steps, opening a box means being awake for a time step, and the presents represent time steps when important data is broadcast)