University of New Mexico
Department of Computer Science

Midterm Examination

CS 561 Data Structures and Algorithms
Fall, 2013

Name:
Email:

- This exam lasts 75 minutes. It is closed book and notes, and no electronic devices are permitted. However, you are allowed to use a 1 page "cheat sheet"
- *Show your work!* You will not get full credit if we cannot figure out how you arrived at your answer.
- Write your solution in the space provided for the corresponding problem.
- If any question is unclear, ask for clarification.

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

1. **Recurrence Relations and Asymptotics**

   In the following, assume as usual, that $f(n)$ is $\theta(1)$ for constant values of n.

   (a) (4 points) The time for Walt to make n pounds of product obeys the following recurrence. $f(n) = 3f(n/2) + n^2$. How much time does it take asymptotically to make n pounds of product?

   (b) (4 points) Jesse has a scheme for recruiting customers with the following properties. The number of customers at day n is twice the number of customers at day $n-1$ minus the number of customers at day $n-2$. Let $f(n)$ be the number of customers at day n. Then we have $f(n) = 2f(n-1) - f(n-2)$. What is the asymptotic value of $f(n)$?

(c) (4 points) Jesse has a new scheme where the number of customers at day n is given by the recurrence $f(n) = 4f(n-1) - 4f(n-2)$. What is the asymptotic value of $f(n)$ now? Will this scheme result in an asymptotically larger number of customers than the scheme from problem 1 (b)?

(d) (8 points) Walt is producing product at an exponentially rate - at day n he produces 2^n pounds. By day n, Jesse can sell 1/2 of the product available at day $n-1$. Thus, the pounds of product that must be stored at day n is given by the following recurrence: $f(n) = 2^n + (1/2)f(n-1)$, with $f(1) = 0$. Prove via induction that $f(n) \leq c2^n$ for some constant c. Solve for the smallest value of c.

2. **Probability and Expectation**

The process for synthesizing a certain chemical can be viewed as a tree over n nodes. In particular, the leaf nodes represent input chemicals, and each internal node represents the process of combining chemicals from its children. During the synthesization process, there is a mistake made at each node (including the leaf nodes) independently with some small probability p.

(a) (5 points) Assume that the impurity of the final result increases by 1 unit for each node at which a mistake is made. What is the expected impurity of the result?

(b) (5 points) Using Markov's inequality, bound the probability that the impurity is at least 2 units.

(c) (10 points) Now the synthesis is performed with more careful checking. In particular, impurity increases by 1 only when a mistake is made at *both* endpoints of an edge in the tree. Now what is the expected impurity of the result?

3. **Amortized Analysis**

Walt is making a device for his friend Hector that will count how many times Hector rings a bell. The software for the device requires a binary counter data structure with INCREMENT and RESET operators.

Recall that in class we discussed an INCREMENT algorithm for incrementing a binary counter in $O(1)$ amortized time. Now we want to also support a RESET operation that sets all the bits in the counter to 0. Below are the algorithms for INCREMENT and RESET. They make use of an array B of bits and a variable m which is the most significant bit.

Algorithm 1 INCREMENT($B[0,\ldots,\infty], m$)

$i \leftarrow 0$
while $B[i] = 1$ **do**
  $B[i] \leftarrow 0$
  $i \leftarrow i + 1$
end while
$B[i] \leftarrow 1$
if $i > m$ **then**
  $m \leftarrow i$
end if

Algorithm 2 RESET($B[0,\ldots,\infty], m$)

for $i \leftarrow 0$ to m **do**
  $B[i] \leftarrow 0$
end for

In the following questions, let n be the number of operations on this binary counter.

(a) (4 points) What is the worst-case run time of INCREMENT as a function of n?

(b) (4 points) What is the worst-case run time of RESET as a function of n?

(c) (12 points) Prove that in an arbitrary sequence of calls to INCREMENT and RESET, each operation has an amortized cost of O(1).

3. **Amortized Analysis, continued.**

4. **Dynamic Programming**

Gus wants to open franchises of his restaurant, *Los Pollos Hermanos*, along Central Avenue. There are n possible locations for franchises, where location i is at mile i on Central. Each location $i > 1$, is thus a distance of 1 mile from the previous one. There are two rules.

- At each location, there can be at most one restaurant, and the profit of a restaurant at location is i is p_i.
- Any two restaurants must be at least 2 miles apart.

(a) (3 points) Jesse proposes the following greedy algorithm for this problem. First, sort the locations by decreasing p_i values, then greedily choose the next possible location, provided that it doesn't conflict with previously chosen locations. Show that Jesse's algorithm doesn't always give maximum profit.

(b) (7 points) Now consider a dynamic programming approach to this problem. For $i \geq 0$, let $m(i)$ be the maximum profit by using locations 1 through i. Write a recurrence relation for $m(i)$. Don't forget the base case(s).

(c) (3 points) In 1-2 sentences, describe how you would create a dynamic program using the previous recurrence. What is the run time of your algorithm?

(d) (7 points) Now Gus wants to solve a generalization of the problem. There are two changes. First, for $1 < i \leq n$, location i is now distance d_i from location $i - 1$. Second, any two restaurants must now be distance k apart for some parameter k. Write a new recurrence relation for this problem. Don't forget the base case(s).

5. **Cartel**

The Juarez cartel is being investigated by the Federales. The n members of the cartel are organized in a perfect binary tree (recall that in a perfect binary tree, all leaves are at the same depth and each internal node has two children). Each day, the Federales select a node uniformly at random from the tree. They then arrest this node and all nodes below it - subordinates are arrested due to incriminating information obtained from the selected node. All arrested nodes are removed from the tree.

(20 points) What is the expected number of days until all nodes are removed? Hint: For each node v, let X_v be an indicator random variable that is 1 if v is arrested *before* any of its ancestors and 0 otherwise. Use linearity of expectation. Your solution can use θ notation.