CS3933 Final Exam
December 14, 2006
Out of 60 marks.

Name:

ID:

Answer all questions in the space provided. Use the back of the page if you need more space. Show your work. No calculators or other aids are allowed. Allowed Time: 3 hours.

1. **(6 points)**
   Algorithm *Spin*(n):

```
x ← 0
for i from 1 to n do
    for j from i to n do
        for k from 1 to i do
            x ← x + (j · k)
return x
```

   Fully analyse the running time of this algorithm asymptotically.

2. **(6 points)** Consider a Divide and Conquer algorithm that runs as follows:

- for input of size n, it produces 4 overlapping parts, each of size $\frac{n}{2}$
- these 4 parts are found in constant time
- it recurses on the 4 parts
- it merges the results of the parts in $c \cdot n$ time, for some constant c (so the merge is in linear time)
- It will not recurse if $n = 1$

This algorithm's running time can be described by the recurrence:

$$T(n) = \begin{cases} 0 & \text{if n=1} \\ 4 \cdot T(\frac{n}{2}) + c \cdot n & \text{otherwise} \end{cases}$$

Complete the asymptotic analysis of this algorithm by finding a closed, nonrecursive, form for $T(n)$ by analysing the recursion tree of the algorithm. You can assume that n is a power of 2.

3. **(9 points)** Design an algorithm that, given an undirected graph $G = (V, E)$ with edge weights $\{w_e \mid e \in E\}$, finds its *densest* component. The density of a component is its total edge weight divided by its number of vertices.

   Analyse the running time of your algorithm.

4. **(9 points)** You have been trying to shop for gifts, and have decided on a set of n specific items. You have also decided to buy one gift per day for the next n days (starting tomorrow).

   Each gift i has a current price p_i, which is the price of that item today. However, all gifts have prices that change in a linear fashion with slope s_i, so that on day j, gift i will cost $p_i + s_i \cdot j$ (where today will be day 0, so gifts are bought on days 1 through n). Some slopes may be negative, but we must still buy one gift per day for the next n days. Do not worry about prices becoming negative. We can assume that all p_i and s_i values are distinct.

   We want to find the order of gifts that minimises the total cost of the gifts; so we want to find a permutation of $1, 2, \ldots, n$, given by $o_1, o_2, \ldots, o_n$ (where gift i is bought on day o_i) that minimises

$$\sum_{i=1}^{n} (p_i + s_i \cdot o_i)$$

   (a) **(4 points)** One potential greedy approach would be to buy the currently most expensive gift first, and continue in descending order of p_i. Give a counterexample and demonstrate why this approach will not always work.

   (b) **(5 points)** Prove that buying the gift with the highest slope first, and continuing in descending order of s_i, will find an optimum order.

5. **(5 points)** For a given sequence of n positive integers a_i, consider the following recurrence:

$$M(i,j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i > j \\ \max_{i \le k < j}\{M(i,k) + M(k+1,j) + a_i \cdot a_{k+1} \cdot a_j\} & \text{otherwise} \end{cases}$$

Design a dynamic programming algorithm that will, given n and the a_i values, compute the value of $M(1,n)$ according to this recurrence. Your algorithm should run in $O(n^3)$ time.

6. **(10 points)** We are planning to walk through a park whose trails are laid out as a grid of n North-South trails and m East-West trails, all intersecting. Each intersection (i, j) has a corresponding altitude $a_{(i,j)}$. We need to travel from the Northwestern corner $(1, 1)$ to the Southeastern corner (n, m), and will only travel East and South. However, there are many paths through these trails that preserve this requirement.

   We want to find a path through this park, traveling only East and South, that will minimise the total altitude climbed up (ignoring any climbing down). If we move from (i, j) to its Eastern neighbour $(i, j + 1)$, we would add the difference in altitude $a_{(i,j+1)} - a_{(i,j)}$ if and only if this value is positive. Similarly, if we move from (i, j) to its Southern neighbour $(i + 1, j)$, we would add the difference in altitude $a_{(i+1,j)} - a_{(i,j)}$ if and only if it is positive.

   According to this measurement scheme, if we have a path of altitudes $6, 3, 10, 7, 4, 8$, it would have a total climbed altitude of $(10 - 3) + (8 - 4) = 11$.

   Design a dynamic programming algorithm that will, given n, m, and the $n \cdot m$ altitude values $a_{(i,j)}$, find the path through the grid that minimises the total altitude climbed.

   Remember to find both the minimum altitude climbed and the path that produces it.

   Analyse the running time of your algorithm.

Additional space for your answer to question 6.

7. **(5 points)** Let X, Y, and Z be three decision problems. Suppose we know that X polynomially reduces to Y and Y polynomially reduces to Z. We also know that Y is in NP. Which of the following can we infer? Briefly explain your answers.

   (a) If Z is NP-complete then so is X.

   (b) If X is NP-complete then so is Y.

   (c) If X is in P, then Z is in P.

   (d) If Z is in P, then Y is in P.

   (e) X is in NP.

8. **(10 points)** Consider the following set problem, known as HITTING SET.
Given: a collection $\mathcal{C} = \{C_i\}$ where each C_i is a set.
Question: Is there a set S, where $|S| \leq k$, such that S intersects all members of $\mathcal{C}$?

$$\forall C_i \in \mathcal{C} \quad S \cap C_i \neq \emptyset$$

Prove that HITTING SET is NP-complete, using the following steps:

(a) **(3 points)** Show that HITTING SET is in NP.

(b) **(7 points)** Show that the graph problem DOMINATING SET $\leq_p$ HITTING SET.