CS 410/586: Test 2, 12 June 2008 Name:_____

You may use the course text and course materials. You may not use computers, other books or other materials. Work individually.

100 points possible. Give your answers on these sheets. (You can ask for extra blank sheets if needed. Put your name on any extra sheets.)

1. String matching (15 points): Describe an efficient method to find the longest string of consecutive a's in a given text.
Hint: How would you search for a string of 50 consecutive a's?

2. Shortest Path (10 points): When a graph has a negative-weight cycle, then clearly Dijkstra's shortest-path algorithm won't work. Consider a graph with negative-weight edges, but no negative-weight cycle. Show that Dijkstra's algorithm could still return incorrect results. Where does the correctness argument for the algorithm break down?

3. Matrices (10 points): Let $M(n)$ be the time to multiply $n \times n$ matrices. Let $S(n)$ be the time to square an $n \times n$ matrix. Show that an $M(n)$-time matrix-multiplication algorithm implies an $O(M(n))$-time squaring algorithm, and that an $S(n)$-time squaring algorithm implies an $O(S(n))$-time matrix multiplication algorithm.

4. (0 points) Give an O(1) algorithm for disjoint-set intersection.

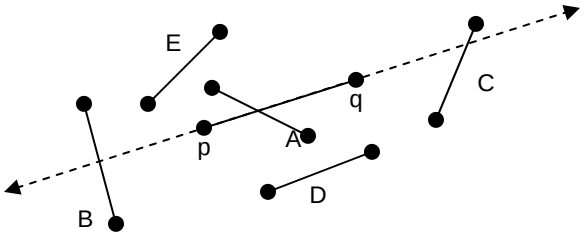
5. Matrices: Give an O(n)-space representation for $n \times n$ permutation matrices such that two permutation matrices can be multiplied in O(n) time, and a regular matrix can be multiplied by a permutation matrix in O(n^2) time. Illustrate your method on the matrices below.

$$P1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$P2 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

$$A = \begin{pmatrix} 1 & 4 & 3 & 9 \\ 2 & 6 & 5 & 8 \\ 2 & 2 & 5 & 8 \\ 3 & 4 & 3 & 1 \end{pmatrix}$$

6. Computational Geometry:



a (10 points): Describe a method to decide if a line segment S crosses the infinite ***line*** defined by two points *p* and *q*. In the figure above, segments A, B and C cross the *p–q* line.
(You can assume that no three endpoints involved are co-linear.) Try to avoid division.

b (10 points): Describe a method to decide if a line segment S crosses the infinite ***ray*** from point *p* through point *q*. In the figure above, only segments A and C cross the *p–q* ray. (You can assume that no three endpoints involved are co-linear.) Try to avoid division.

7. NP-completeness (15 points): Undirected Graph G1 is *isomorphic* to graph G2 if we can rename the vertices in G1 to obtain G2. The *subgraph isomorphism problem* is: Given undirected graphs G and H, determine if graph G is isomorphic to a subgraph of H. Show that the subgraph isomorphism problem is NP-complete.

8. NP approximation (10 points): Suppose A is an approximation algorithm that can find a vertex cover of an undirected graph that is no more than 2 off from the optimum. That is, if G is a graph with a vertex cover N1 of size k, then A will find a vertex cover N2 for G of size at most $k + 2$.

Show that if A has polynomial time complexity, then the vertex-cover problem can be solved in polynomial time.