

Midterm 2

CSC 282

8 December 2005

*Write your **NAME** legibly on the bluebook. Work all problems, but notice selections or extra credit points within problems. No book. One page (letter size, both sides) of notes OK. Please turn in notes with test.*

1. FFT: 15 Min. + 5 extra credit

1.A: (15 points):

You want to perform a convolution of an $m \times m$ template (array of real numbers) with an $n \times n$ image (array of real numbers). m and n are powers of two. What relative sizes of m and n make it cheaper to use the convolution theorem and the transform domain than it is just to perform the convolution by the “shift, multiply and add” method? Or if you find it easier, just let $n = 1028$ and say how big m has to be.

1.B: (5 points extra credit)

The Fourier transform of a real function is symmetrical, and the Fourier transform of a symmetric function is real. Say why.

2. Max Flow: 15 Min.

My two kids are going through a phase where they just won't have anything to do with each other. They will not even walk down the same block, even at different times, to get to school. For example, if one walks down D street between 3rd and 4th Avenue, the other one won't. Luckily, it's OK if their paths cross at a corner; also luckily our house and the school are both on a corner. And luckily I have a map of the city and have just purchased Apple's new OS-X MaxFlo(TM) application. Can I set this situation up as a maximum flow problem to see if my kids can get to school? If so, how?

3. LP: 15 Min. + 5 extra credit

3.A: (15 points): Solve this LP problem by the Simplex method, explaining your work as you go.

Minimize $y - x$

Subject to

$$-6y - x \geq -12$$

$$x, y \geq 0$$

3.B: (5 points extra credit)

Solve the above Linear Program graphically.

4. Graphs, Traversals, Paths: 15 Min.

Do both parts.

4.A:

Consider the problem of testing, given a connected, edge-weighted, undirected graph $G = (V, E)$ and an integer M , whether G has a spanning tree whose largest individual edge weight is at most M . Design an algorithm for solving this problem in time $O(|E| + |V|)$.

4.B:

Consider the special case of the single-source shortest path problem in which the edge weights are either 1 or 2. Show that this special case can be solved in time $O(|V| + |E|)$. Hint: What if a graph only has weights of 1?

5. CG: 15 Min.

Do part 5.A and choose one of 5.B or 5.C.

5.A. (Do this) Give an $O(n \lg n)$ time algorithm to determine whether two simple (closed, non-self-intersecting) polygons intersect. (This is easy: one sentence is enough.)

Do one of 5.B or 5.C:

5.B. Given a set of points S , prove that the pair of points farthest from each other must be vertices of the convex hull of S .

5.C. (There are two parts to this one.)

5.C.1 In the spirit of our Computational Geometry chapter (hint), how can you compute the area of an arbitrary triangle as shown below with three applications of a vector operation? (The figure is also meant to be a hint.)

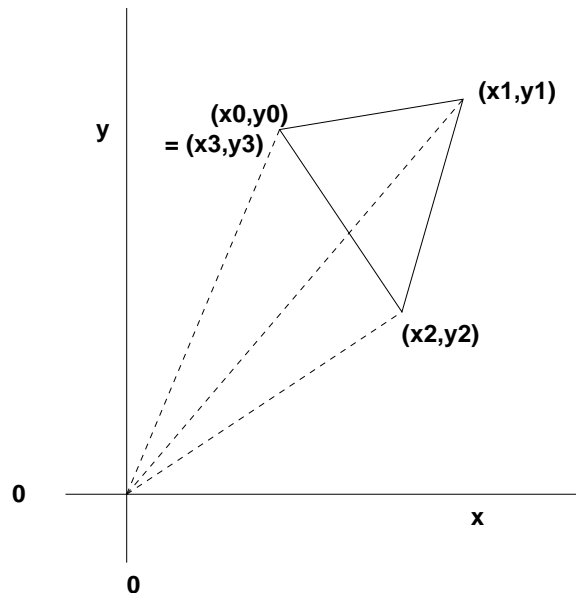


Figure 1: An n -point polygon with $n = 3$. One vertex has two names.

5.C.2 With the answer to 5.C.1 as a start, show how to compute the area of an n -vertex, simple, not necessarily convex polygon in $\Theta(n)$ time. A closed-form answer would be nice (hint: it's a sum from $i = 0$ to $i = n$.) A proof of correctness would also be good.