## Programming Assignment 1

## due Nov. 6, 2012

The basic task is this: given a set of points, return the set of lines containing at least 4 of the points.

The points will be provided as space-separated pairs of values in decimal notation on separate lines in a file 'points.dat'. Treat the pairs of real numbers provided as pairs of coordinates, first the x-coordinate, then the y-coordinate. You are not responsible for error checking on the file.

The return should be output to a file in lines where each line contains the slope and y-intercept of a line containing at least 4 points of the input data. The values should also be printed to the screen. (Recall that a line may be defined by an equation y = mx + b, where m is the slope of the line and b is the y-intercept. Given two points,  $(a_1, b_1)$  and  $(a_2, b_2)$ , the slope of the line joining them is  $m = \frac{b_2 - b_1}{a_2 - a_1}$ . The intercept is  $b_i - ma_i$ .)

Due to floating point approximations, the slope and intercept calculated from different pairs of points on the same line may be slightly different. For the purposes of this assignment, you may treat values that differ by less than  $10^{-8}$  as equal.

Please hand in your code by committing it to the your course svn repository. Basic information about svn can be obtained at https://svn.cs.du.edu. The course repository is https://svn.cs.du.edu/courses/comp2370/f2012/

The code should be well-organized and well-documented. Output should be user-friendly. For example, each line containing 4 or more points should appear exactly once in the output. Further, the user should be informed of the convention for reporting vertical lines.

Please prepare a description of your algorithm and a calculation of the  $\Theta$ -bound for its worst case running time. It is hoped that your algorithm will be  $o(n^4)$ , the running time for an exhaustive search of all quadruples of points being  $\Theta(n^4)$ . This work is due at the beginning of class on Monday, November 5 for the MW section, and at the beginning of class on Tuesday, November 6 for the TuTh section.

You may work individually or in pairs. A pair may hand in a single program, but each member of the pair should hand in a separate description and analysis of the algorithm.