CS 410/586: Quiz 9, 1 June 2009 Name:____KEY_____

No books or notes. Work individually.

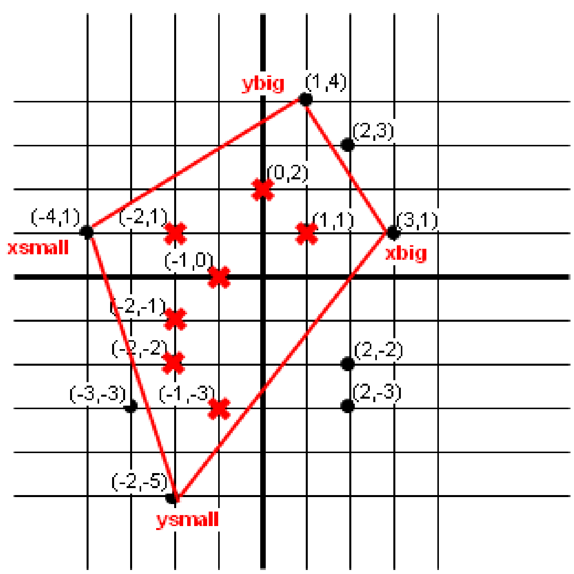
The time complexity of convex-hull algorithms depends on the number of points in the input. Being able to quickly "prune" points from the input that can't be on the hull might reduce the total time.

9-A (5 points) Consider the following pruning method. Given a set of points **Q** for which we want to compute the convex hull, find the following points (breaking ties arbitrarily):

xbig: point with the greatest x-coordinate
ybig: point with the greatest y-coordinate
xsmall: point with the least x-coordinate
ysmall: point with the least y-coordinate

Form a polygon **P** from the points *xbig*, *ybig*, *xsmall*, *ysmall*. Discard all points from **Q** that are inside **P**.

For the set of points **Q** below, draw **P**, and cross out (×) all the points discarded by this method.



9-B (5 points) Is the method from 9-A always guaranteed to discard some points from **Q**? Why or why not?

There are many situations where the method will remove no points, among them:

- *Polygon P is degenerate because xbig = ybig and xsmall = ysmall.*
- *All points are on the convex hull*
- *There are only four points to begin with*
- *The set Q is the result of applying the method. (Applying the method a second time removes no additional points.)*