

Analysis of Algorithms

Programming Project 1 – Divide and Conquer

In this project, we will compute the convex hull of a given set of n points in two dimensions (i.e., each point has an x-coordinate and a y-coordinate) using a Divide and Conquer algorithm. We discussed this problem and the algorithm in Lecture 6 as well as in the Programming Project 1 overview, so if you are unfamiliar with the general problem, please consult those lectures first.

Stating the parameters formally:

- The input files we will test your code on will all have at least 3 points (and as many as 1 million points) that are sorted by x-coordinate. No two points will have the exact same x-coordinate and no three points will lie on the exact same line (which helps with you not needing to consider some edge cases in the algorithm).
- The indices of the points from the input file should start from 0. So the first point in the input file is point 0, the next point is point 1, and then last point is point $n-1$.
- Your output should be to print a list of the indices of the points on the convex hull of the input point set to the file 'output.txt', one index on its own line (see the example output file for an example). In class we assumed that you would list the points in counterclockwise order (the pseudocode that we gave to compute the tangents uses this assumption), but your output can print the points in either clockwise or counterclockwise order. Also, it doesn't matter which point you start from so long as the list has the correct points in the correct order.

If you do not give a divide and conquer algorithm for this project, you will get a 0. The entire point of this project is to design a divide and conquer algorithm, not just to get some solution that works.

How to code this project:

For the most part, you can code this project in any language you want provided that the language is supported on the Fox servers at UTSA. Most students in the past code in either C, Java, or Python. We want you to use a language that you are very comfortable with so that implementation issues do not prevent you from accomplishing the project. That said, we will be compiling and running your code on the Fox servers, so you need to pick a language that is already installed on those machines. See here for more information on how to connect to the Fox servers remotely (also covered in the Programming Project 1 overview lecture): <https://caicc.utsa.edu/computer-science/resources.html> under the tab "Main Lab Frequently Asked Questions".

If you are coding in Java using Eclipse, you can submit an Eclipse project! You still must make sure that it complies on the Fox servers! You can do this through VDI. To export/import an Eclipse project, see this nice YouTube video:

<https://www.youtube.com/watch?v=NMfIHuEwE7A>

For folks compiling at the command line, we are asking you to provide a bash script named *project1.sh* that will act similarly to a makefile. In short, your bash script is basically just a text file with compile/execute your code. If coding in C, it would be two lines: (1) the command to compile your code, and then on a different line, (2) it should contain the line to execute your code. If coding in Python, since it isn't compiled, it should simply contain the line to execute your code.

Files provided in the project:

Since you are programming in different languages, we are providing no source files for your project. Here are the files that are included in the project:

- 1) *input.csv* - This file contains the input points in a comma separated values format. Basically, each line contains both coordinates for 1 point in the following format: x-coordinate,y-coordinate. In particular, the comma separates the end of the x-coordinate from the start of the y-coordinate, so use the comma to split the line into coordinates for each point.
- 2) *output.txt* - This file contains a sample output for the given input file. Your code should generate this file inside of your program. The contents of the output file are the indices of the points of the convex hull for the points in *input.csv* listed in counterclockwise order.
- 3) *DisplaySolution.py* - This is a short python program that will draw the points from *input.csv*, and if the *output.txt* file exists will draw the convex hull as well. I have provided this program to help you debug. Your program doesn't actually need to do anything with this file, although you can see an example of how I read in the input file which might help you in your own project.
- 4) *project1.sh* - a blank bash script file for those of you who are not submitting an Eclipse project. If coding in C, the first line should be the line to compile the code and the second line should be the line to execute the code. If coding in Python, it should simply be the line to execute your code.
- 5) *NamesOfStudentsForThisSubmission.txt* – If you are working with a partner, please write both partner's names in this file.

Grading

We will grade according to the rubric provided on Canvas. The majority of the points come from the following: did the student give a divide and conquer algorithm that returns the correct answer for all possible inputs? Proper documentation may help the grader understand your code and earn you partial credit in the event you have some mistakes in the code.

Violations of the UTSA Student Code of Conduct will be penalized harshly. In particular, be very careful about sending code to a student who asks how you accomplished a particular task. I've heard this story several times recently: "They said they just wanted to see how to perform part X of the project. I didn't think they would submit my exact code." If this happens, you will both be penalized for cheating. To protect yourself and to more properly help your fellow student, send pseudocode, and not actual completable code.

Also we know about the online sites where people upload projects and have a third party complete the project for you. This is a particularly egregious form of cheating (it's in the best interest of your career to not tolerate this). If you use a solution from one of these sites or submit a minor modification (minor is at the discretion of the instructor) of a solution from one of these sites, you will receive a 0 and will be reported to the university for a violation of the UTSA Student Code of Conduct.

Submitting

Please put your name (and your partners name if you have a partner) in the file *NamesOfStudentsForThisSubmission.txt*.

Zip up your project folder (including all of the input files) and submit on the dropbox on Canvas by the due date.