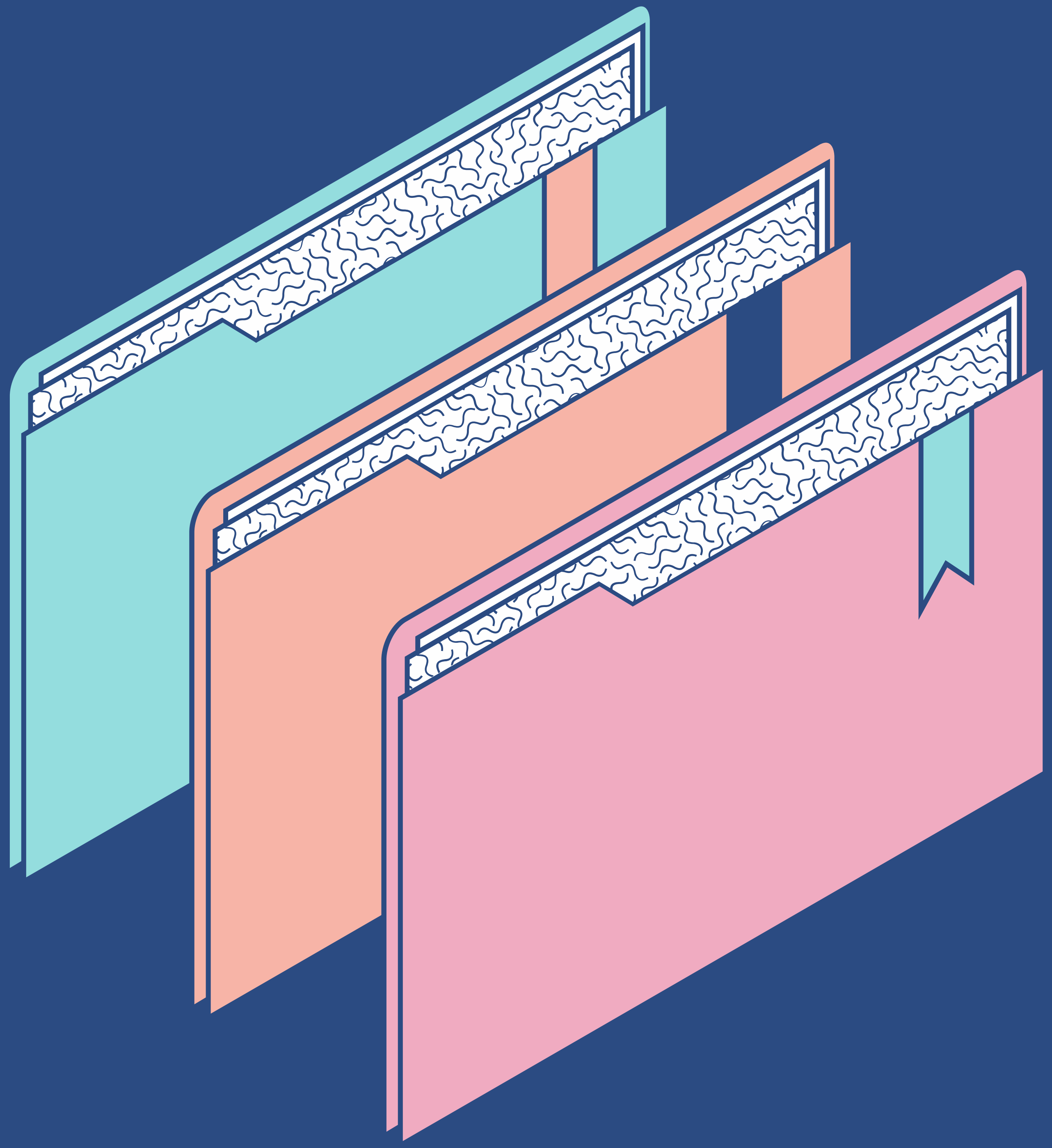
INDIVIDUAL COMPUTATIONAL GEOMETRY PROJECT

Closest-Pair

Vraja Mihai Alexandru Ovidiu



The Problem

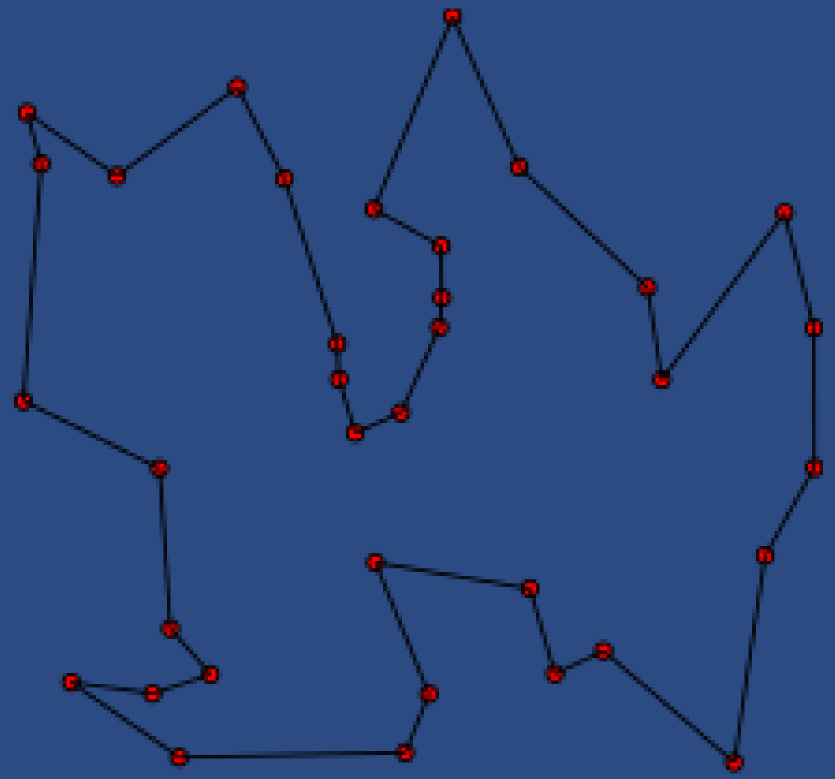
When given a set of points, we must compute a pair whose distance is smaller than(or equal) to all other distances between points.

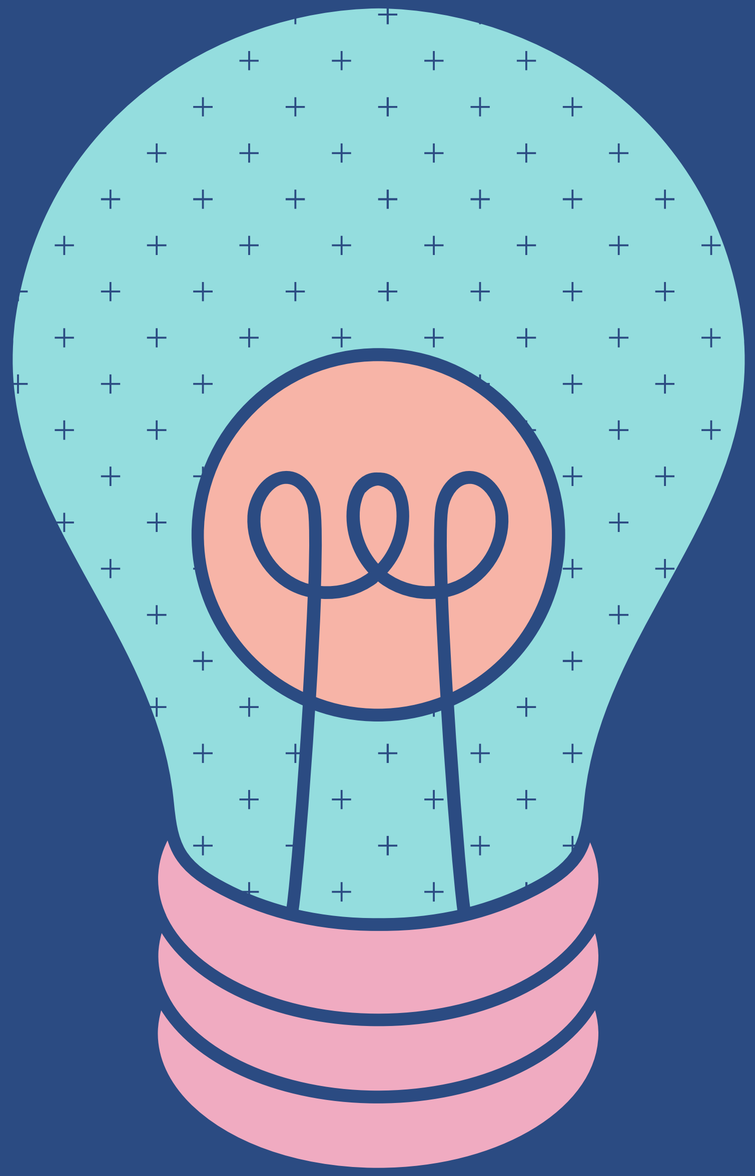
It must be at least n*log(n) efficient.

Why this project?

The closest pair project was arguably the most bare-bones one, but that’s not neccesarly a negative.

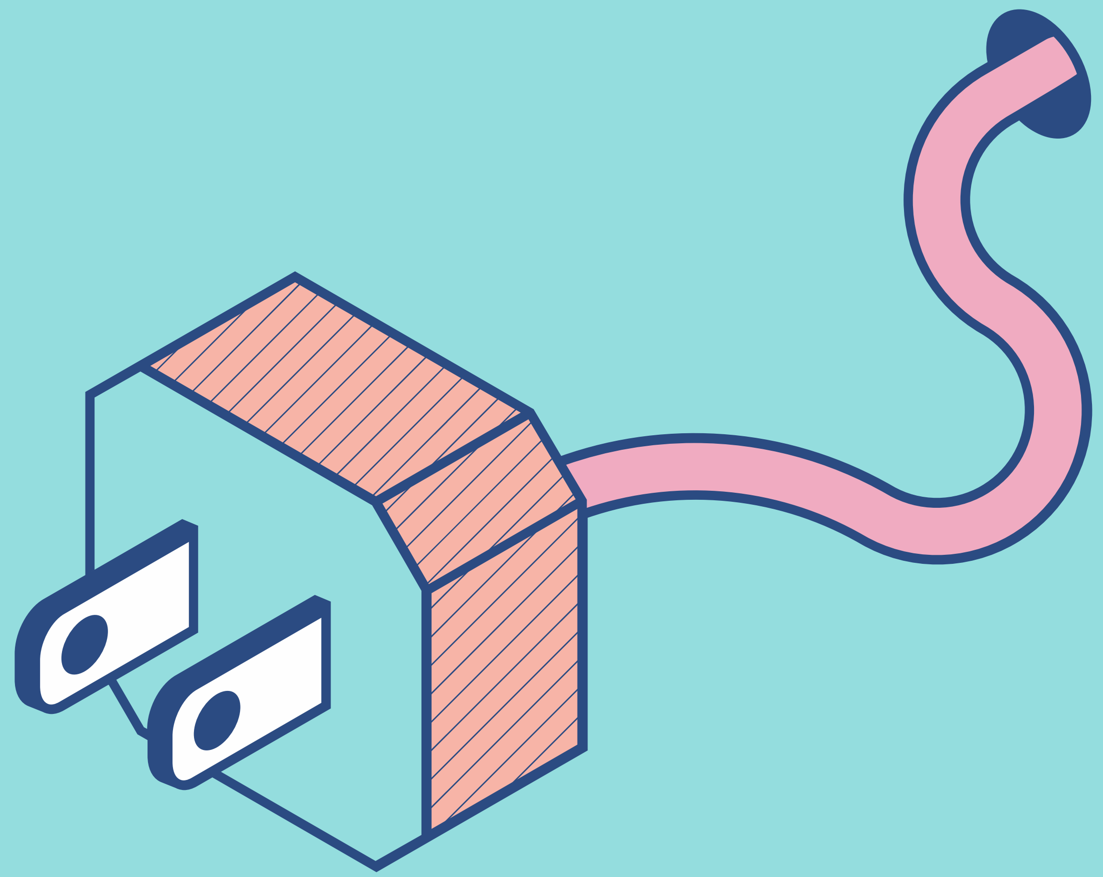
The simplistic nature of it makes it so that it is relevant in a wide variety of other projects and problems.





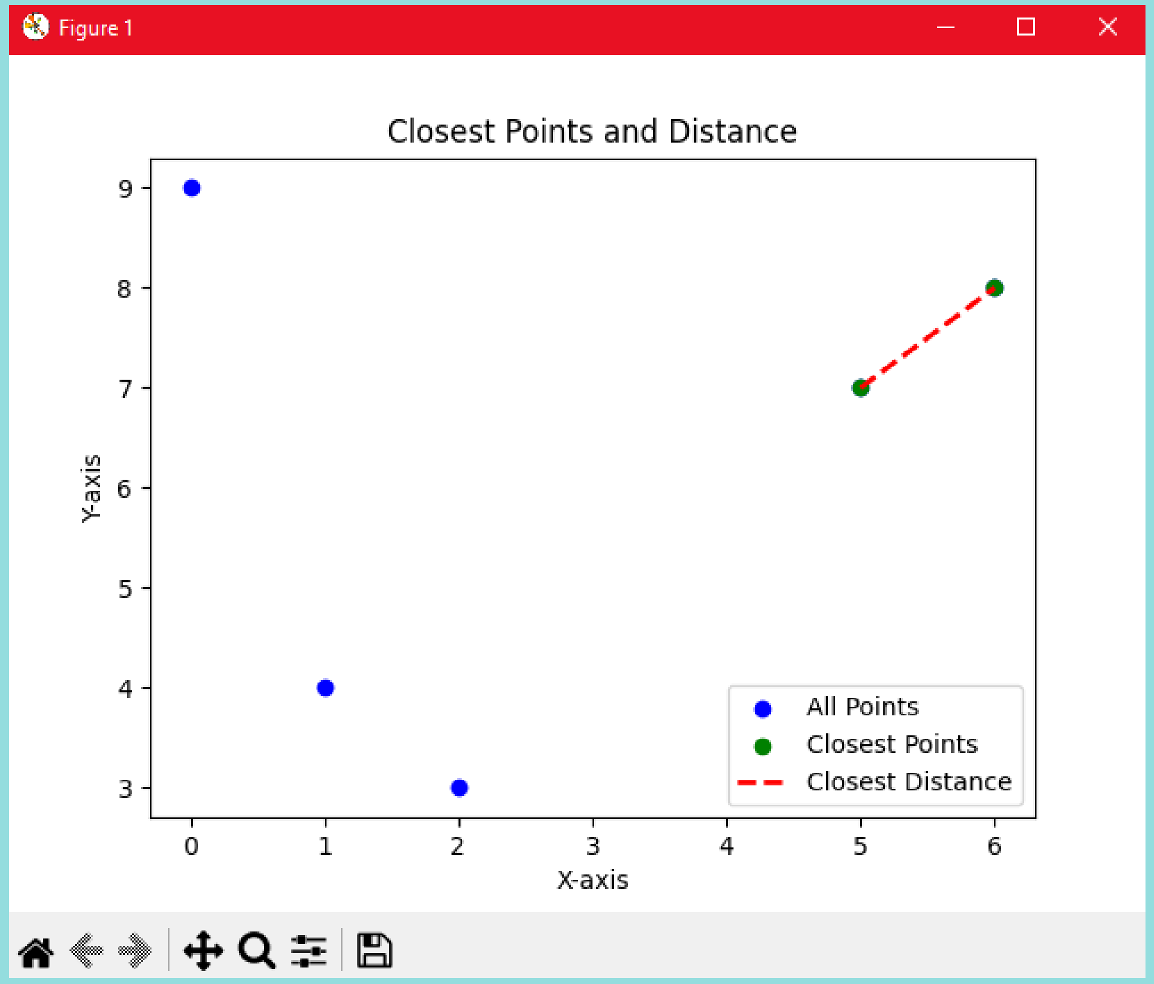
The Result

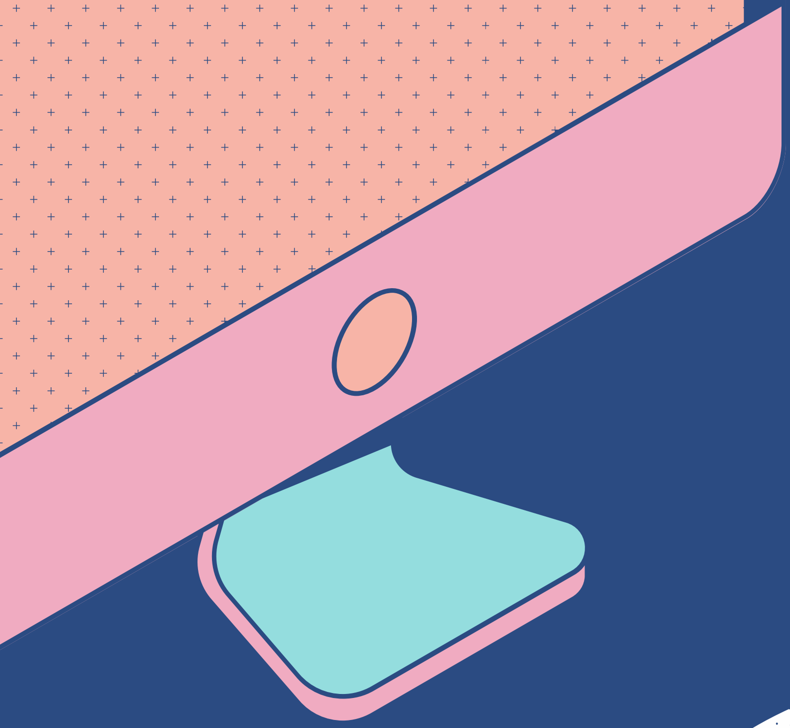
ClosestPairPy is a Python program designed to find the pair of closest points among a given set of 2D points. The algorithm employs a divide-and-conquer approach to efficiently compute the closest points. Additionally, the program includes a visualization feature using Matplotlib to display the input points and the identified closest pair.

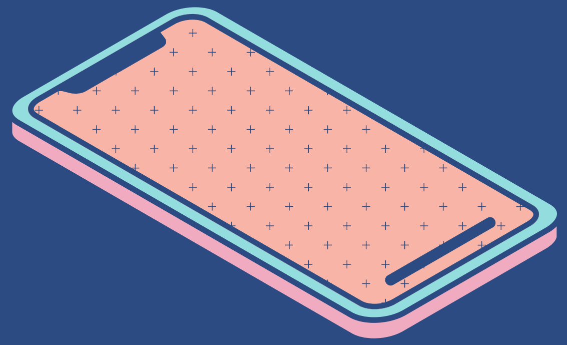


How it works

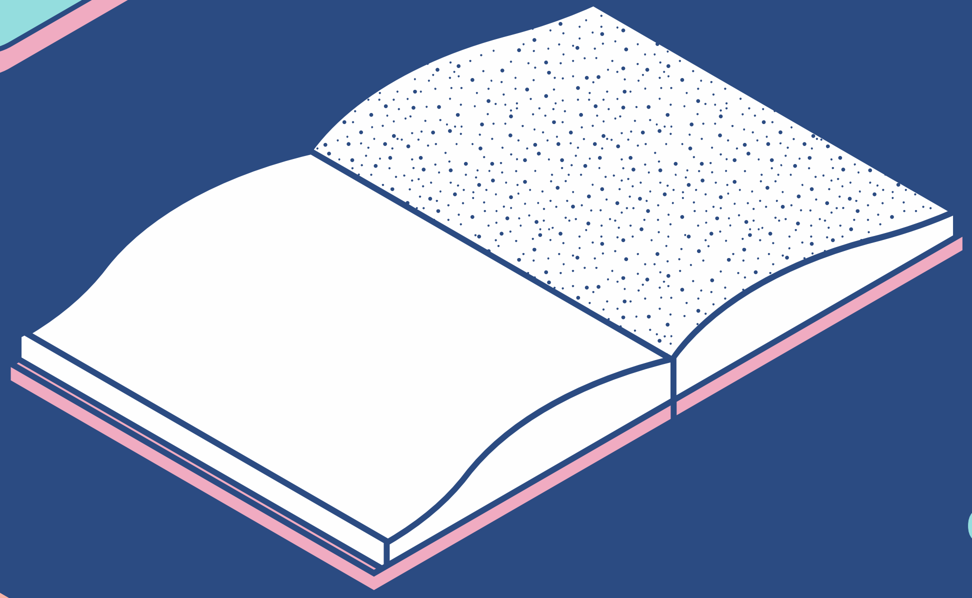
1. Run the script in a Python environment.
2. Enter the number of points when prompted.
3. Input the x and y coordinates for each point.
4. The program will output the closest points and their distance.
5. A graphical representation of the points and the closest pair will be displayed.







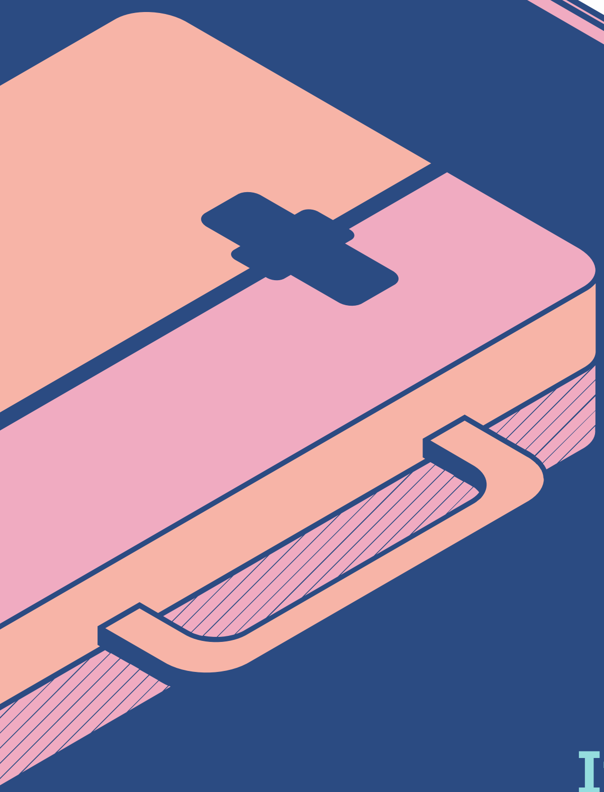
How it operates



The program uses the following functions to ensure correct and efficient results:

# calculate_distance(point1, point2)
Calculates the Euclidean distance between two points in a 2D plane.

# find_closest_points(points)
Finds the closest pair of points from a given set of 2D points using a divide-and-conquer algorithm.



# closest_pair_recursive(points)
Recursive function used within `find_closest_points` to find the closest pair efficiently.

# closest_pair_in_strip(strip, d)
Helper function to find the closest pair within a specified strip of points.

# plot_points(points, closest_points)
Generates a scatter plot using Matplotlib to visualize the input points and the identified closest pair.

It also has the # Main Code
The main code initializes the program, collects user input for points, and executes the core functionalities.

The program is also reliant on the matplotlib dependency.

Thank you for your time!