# Implementing Three Voronoi Diagram Computation Algorithms and Comparing Their Performance

Barkın Saday - 21902967

February 2025

## Project Description

This project aims to implement a software application that computes and visualizes Voronoi Diagrams in 2D using three distinct algorithms: the Randomized Incremental Algorithm, Fortune's Algorithm (sweepline approach), and the Flipping Algorithm. The objective is to compare the performance of these algorithms in terms of execution time and efficiency across different input sizes, ranging from 100 to 1,000,000 random points generated using various distributions.

The program will feature a user interface that allows users to specify parameters such as the number of points and provides interactive tools for zooming and translating the visualization of the Voronoi Diagram. The application will also visualize the step-by-step execution of Fortune's Algorithm, offering an educational insight into its operation.

Performance comparisons will be conducted using a reasonable set of test cases, and the results will be documented in a final report, highlighting the strengths and weaknesses of each approach.

## Technologies and Tools to be Used ()

- **Programming Language:** Python
- **Computational Geometry Libraries:**
  - **SciPy:** For Delaunay Triangulation and geometry processing
  - **Shapely:** For geometric operations and handling Voronoi cells
- **Visualization:**
  - **Matplotlib:** To visualize the Voronoi Diagrams and provide interactive features
  - **PyQt or PySide:** For creating a graphical user interface (GUI) with zoom and pan capabilities
- **Random Point Generation:**
  - **NumPy:** For generating random points with various distributions
- **Performance Measurement:**
  - **time module (Python):** To measure and compare the execution time of the implemented algorithms