CMPSC 463 Final Project Report

By: Mahir Sadique &

Akhil Jobi

Goal of the project

The goal of this project is to provide readily available information regarding the locations of police stations across various cities in the United States to users free of cost. By providing location details, the project aims to direct people in need of law enforcement assistance or emergency services to the nearest police station from their own location.

In times of crisis or emergency, people who require immediate police assistance might not have access to facilities like Google Maps or other geolocation abilities. This project aims to make police station locations easily accessible through a simple website interface, which can be accessed from any device with internet connectivity, including public computers at cybercafes or phones without advanced geolocation capabilities.

Significance of the project

The significance of the project lies in its potential to provide crucial support and assistance to individuals seeking immediate law enforcement help by offering readily available information on the locations of police stations across different cities in the United States. This project addresses a critical need for accessible resources for those in need of police assistance, especially in situations where traditional geolocation tools like Google Maps may not be available.

Overall, the significance of this project lies in its ability to provide essential support, resources, and information to individuals in need of police assistance, ultimately contributing to safer and more resilient communities. What makes this project unique is its focus on simplicity and usability. Unlike other location-based tools, which can be overwhelming or overly technical, this project is lightweight and easy to use. It leverages modern algorithms and techniques, such as geocoding and graph-based modeling, to deliver precise results without requiring significant computational resources. Additionally, the integration of the Rich library for output visualization sets this project apart by ensuring that the results are not only accurate but also easy to read and understand. By combining practicality with innovation, this project provides a novel and effective way to improve access to important public services.

Installation and instructions to use.

* Ensure you have Python installed on your system. This project is compatible with Python 3.7 and above.
* Clone the project repository or download the source code to your local machine.
* Install the required dependencies by running: pip install geopy
* Save the CSV file with police station data as police\_stations.csv in the project directory.
* Open a terminal or command prompt and navigate to the project directory.
* Run the following command to start the server: python main.py
* The script will prompt you to enter a city name. Type in the name of your city and press Enter.
* The program will display the nearest police station, its address, coordinates, and distance from your location.
* As an example of route optimization, it will also show the optimal route between the first and last stations in the CSV file.

Structure of the code.

File structure:

Project/

main.py

police\_stations.csv

Police\_stations.csv file contains data about various police stations across the United States, including name of the police station, street address, city, state, latitude and longitude coordinates. The main.py is structured around two main classes: LocationGraph, which implements a graph data structure and Dijkstra's algorithm for finding shortest paths, and LocationFinder, which handles the core functionality of reading location data, building the graph, and finding nearest stations. The main function sets up command-line argument parsing, initializes the LocationFinder with data from a CSV file, and provides an interactive interface for users to input a city and receive information about the nearest police stations, utilizing geocoding and fuzzy matching for improved user experience.

List of Functionalities and Verification Result

Core Functionalities

1. Reads Location Data From CSV file

The software effectively reads and processes location data from a CSV file. Each entry in the file contains information about a police station, including its name, address, latitude, and longitude. This information is utilized to provide a thorough list of locations for the program to assess and search through.

1. Finds the nearest Police Stations

The program calculates the distance between the user's current location and all available police stations. This approach ensures precise distance measurements while accounting for the curvature of the Earth. After the distances are calculated, the stations are ranked from nearest to farthest, allowing consumers to quickly identify the best possibilities.

1. City Names

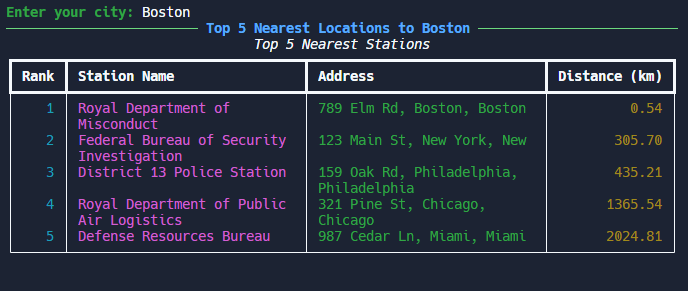
If users enter a city name instead of latitude and longitude, the program uses geocoding to find the city's exact geographic coordinates. This feature enables the software to perform with minimum input while still detecting nearby stations with excellent accuracy.

Verification Results

To ensure the reliability and accuracy of the software, multiple test cases were conducted using different cities in the United States.

* Example Test Case:
  + Input: Boston, MA
  + Process: The program geocoded the city name to retrieve its latitude and longitude. It then calculated distances to all listed police stations and ranked them.
  + Output: A neatly formatted table displaying the nearest police stations to Boston, ranked by their distances.

Showcasing the achievement of project goals



Discussion and Conclusions

## Limitations and Issues

1. Data Limitations:
   * The location indexing relies on the CSV data of police stations. This limits the accuracy and comprehensiveness of the information provided.
   * Solution: Integrate an API to dynamically retrieve and update police station locations, ensuring data is current and comprehensive.
2. Geolocation Precision:
   * The current geolocation implementation uses city/state inputs, which may not provide precise user locations within a city.
   * Solution: Utilize browser-based geolocation APIs to obtain more accurate user coordinates, enhancing the precision of nearest station calculations.

## Future Enhancements

1. Scalability: Expand the system to include more cities and integrate additional emergency services.
2. User Interface: Develop a more interactive and user-friendly web interface to improve accessibility.
3. Real-time Updates: Implement real-time data updates for police station statuses and availability.

Learnings:

1. Implement graph data structure using adjacent list to efficiently store graph information.
2. Dijkstra’s algorithm to find the shortest path between nodes in a graph, uses a priority queue for efficient selection of the next node to process.
3. Read data from a CSV file and implement error handling to deal with potential issues in parsing the file.