Data Structures and Algorithms

CRIMENET

Course Project Report

School of Computer Science and Engineering

2023-24

Contents

1. Course and Team Details

1.1 Course details

Course Name	Data Structures and Algorithms
Course Code	23ECSC205
Semester	III
Division	B
Year	2023-24
Instructor	Dr . PRIYANKA GAVADE

1.2 Team Details

Si. No.	Roll No.	Name
1.	06	ANIKET D PATIL
2.	35	SANDESH ARUN CHAVAN
3.	39	SANJANA S BIRADAR
4.	64	SOUMYA N PATIL

1.3 Report Owner

Roll No.	Name
35	SANDESH ARUN CHAVAN

2. Introduction

The chosen project revolves around the development of a comprehensive software system designed to facilitate various aspects of police station management and law enforcement operations. This project finds its domain within the broader realm of law enforcement technology and public safety. The primary objective is to create a user-friendly and efficient platform that assists in managing police station resources, streamlining information about police employees and their respective designations, providing access to geographic data for efficient deployment of law enforcement resources, and offering public accessibility to critical information, such as the contact details of relevant personnel for various police stations. By integrating advanced technologies and geographic information systems (GIS), the software aims to improve operational efficiency, enhance public safety, and enable law enforcement agencies to respond effectively to incidents and emergencies.

The decision to pursue this project stems from the insights gained from a white paper on the modernization of law enforcement through technology, which highlighted the significance of implementing information systems and GIS for optimizing police operations and resource allocation. The white paper emphasized the importance of creating accessible and responsive platforms to improve public interaction with law enforcement agencies. Consequently, the project selection aligns with the recommendations identified in the white paper, aiming to address the contemporary challenges faced by law enforcement agencies through the application of innovative technological solutions.

3. Problem Statement

3.1 Domain

Problem Statement :
The problem statement revolves around the inefficient management of police station resources and the lack of seamless accessibility to critical information, such as the contact details of relevant personnel, for various police stations. This inefficiency can lead to delays in responding to emergencies, lack of clarity in resource deployment, and difficulties for the public in accessing necessary information. The project aims to automate the process of managing police station resources, streamlining information about police employees and their respective designations, and providing accessible data for efficient deployment of law enforcement resources.

Why selected as the need statement from the white paper :
The need statement was selected from the white paper due to its alignment with critical pain points identified in contemporary law enforcement operations. The white paper emphasized the challenges related to the inefficient management of law enforcement resources and the lack of accessible information for the public. These issues were recognized as significant barriers to effective law enforcement and public safety.

Motivation :
The motivation behind selecting this need statement was driven by the potential impact of addressing these challenges. By automating the management of police station resources and providing accessible information, the project can significantly enhance the efficiency of law enforcement operations, streamline resource allocation, and improve public interaction with law enforcement agencies. Consequently, addressing these needs can lead to tangible benefits in terms of public safety, operational effectiveness, and community engagement.

3.2 Module Description

1. Creation of CSV Files for Police Database and Criminal Database:

To lay the groundwork for efficient data management, the task involves creating separate CSV (Comma-Separated Values) files for the police database and criminal database that cater to individual police stations. This initiative will streamline data organization and facilitate seamless access to critical information for each police station.

2. Insertion of Criminal and Police data into Respective Police Station Databases:

The next step involves adding criminal data into the specific police station's database files while ensuring that the data does not already exist. This process enhances the accuracy and relevance of the criminal data held by each police station, contributing to more effective law enforcement and crime management.

3. Creation of a Graph Using Linked List and Weighing the Edges:

Leveraging data from Google Maps, an intricate graph will be created using a linked list data structure to represent the connectivity and spatial relationships within the entire Belgaum city. Accurate weights will be assigned to the edges of the graph, likely denoting distances or other relevant metrics, to enable precise analysis and applications, such as route optimization or resource allocation.

4. Criminal Search Across All Police Station Files Using KMP String Pattern Matching Algorithm:

The implementation of the Knuth-Morris-Pratt (KMP) string pattern matching algorithm will enable the efficient search for specific criminal records across all police station database files. This advanced algorithm enhances the search capabilities, allowing for swift and accurate identification of pertinent information related to criminal records.

5. Menu System with Police Authentication with hidden Password :

It offers distinct functionalities for the public and police, with password authentication ensuring secure access to police-specific operations. The program hides the entered password. The program allows users to explore city information, find the nearest police station, search for criminals, and perform other relevant tasks.

4. Functionality Selection

Module : Foundations

Si. No.	Functionality Name	Known	Unknown	Principles applicable	Algorithms	Data Structures
1	Creation of CSV Files for Police Database and Criminal Database	The CSV (Comma-Separated - Values) files are used to store the data into table format. The data already known is number of files to be created according to police stations. Criminal data approximately contains criminal id, criminal name, age, gender, crime details, crime date.	Maintaining consistency and accuracy in criminal data, such as ensuring correct formatting, avoiding duplicate entries, and handling updates or corrections, can be challenging in a CSV-based system. Inaccuracies or inconsistencies in criminal data may lead to flawed law enforcement decisions, hinder investigations, and compromise the overall reliability of the system.	Principles : 1. Ensure the accuracy, completeness, and consistency of data within CSV files. 2. Organize data to minimize redundancy and improve efficiency. Design Techniques : 1. Include a header row in the CSV files. 2. Maintain consistent delimiters throughout the CSV files.	-	Arrays, Structures
2	Insertion of Criminal Data into Respective Police Station Databases	Table formatted data is stored in CSV (Comma-Separated-Values) files. Criminal data, which is previously known, roughly consists of the following: age, gender, criminal details, criminal date, criminal id, and criminal name.	In a CSV-based system, it can be difficult to maintain consistency and accuracy in criminal data, including making sure that formatting is proper, preventing duplicate entries, and managing updates or corrections. Errors or discrepancies in criminal data could affect investigations, influence law enforcement choices, and jeopardize the system's general dependability.	Principles : 1. Validate criminal data before insertion. Design Techniques : 1. Insert data in batches rather than individually. 2. Insert data incrementally.	-	Arrays

3	Menu System with Police Authentication with hidden Password	The Public can access the only given options to them. Police can access Public's plus their own options like accessing the criminal database	How to hide the password when the Police authority types the password	Principles : 1.Validate the correct choice. 2.Validate the password.	-	Arrays

Module : Structured Data Management

Si. No.	Functionality Name	Known	Unknown	Principles applicable	Algorithms	Data Structures
1	Creation of a Graph Using Linked List and Weighing the Edges	Graphs, fundamental in computer science, represent relationships with nodes and edges. Directed or undirected, they solve problems like pathfinding efficiently, vital in applications like social networks. Commonly represented using adjacency matrices or lists. The data already known is number of location(nodes) in city.	Representing an entire city in a graph could lead to a large and dense graph, posing challenges in terms of memory usage, processing time, and overall system efficiency. Identifying and selecting appropriate graph traversal algorithms for efficient analysis and applications such as route optimization. This includes considering factors like Dijkstra's algorithm or A* search algorithm. Ensuring the scalability of the system to handle a large city graph efficiently, especially as the city grows or evolves over time.	Principles : 1. Select between adjacency matrix or adjacency list representation, considering factors like memory efficiency, ease of traversal, and the density of the graph. 2. Clearly define the adjacency list for edge weights based on the specific needs of the application. Design Techniques : 1. Implement linked lists for adjacency lists in graph representation.	Dijkstra's	Adjacency list, Adjacency matrix

Module : Sorting and Searching

Si. No.	Functionality Name	Known	Unknown	Principles applicable	Algorithms	Data Structures
1	Criminal Search Across All Police Station Files Using KMP String Pattern Matching Algorithm	The module aims to implement the Knuth-Morris-Pratt (KMP) string pattern matching algorithm for efficient searching of specific criminal records across all police station database files. The already known is it matches the pattern in string and tells it is matched or not The process information regarding how the KMP algorithm integrates with the police station database files	If the police station database is extensive, searching across all files using the KMP algorithm could be computationally expensive and time-consuming. Understanding the structure of the police station database, including the organization of tables, fields, and relationships, is vital for effective implementation. Detailed knowledge about the characteristics of criminal records, including the type of information stored (e.g., names, addresses, criminal activities), helps in designing effective search patterns. Minimizing the occurrence of false positives (incorrect matches) and false negatives (missed matches) by refining the string pattern selection and algorithm parameters.	Principles : 1. Understand and adhere to the structure of CSV files storing criminal records. 2. Implement robust error handling and logging mechanisms. Design Techniques : 1. Create an index for efficient pattern matching. 2. Design a user-friendly interface for input and feedback.	KMP String Pattern Matching	(LPS - Longest Prefix Suffix) Array,

5. Functionality Analysis

5.1 : Creation of CSV Files for Police Database and Criminal Database

5.1.1 Workflow :

Creating CSV files for a Police Database and a Criminal Database involves organizing and formatting data in a structured manner.

- Identified the necessary fields for both the Police Database and Criminal Database. For Example,
  Police database : Police id, Name, Rank, Salary etc
  Criminal database : Criminal id, Name, Age, Gender, Crime details, Crime date etc

- Created CSV file headers
  Created a new file for each database with headers corresponding to the fields identified in Microsoft Excel

- Entered data
  Populated the CSV files with sample data, ensuring each row aligns with the corresponding headers

5.1.2 Efficiency Analysis :

- The time complexity of creating a CSV file is generally linear with respect to the number of records or rows being written. It is O(n), where n is the number of rows.
- Space Complexity of CSV files is S(n).

5.2 : Insertion of Criminal and Police data into Respective Police Station Databases

5.2.1 Workflow :

The insertion of criminal and police data into respective police station databases involves managing and updating information related to both criminals and law enforcement officers.

- Data collected is only a sample data
- Checked if the criminal already exists in the database. If the criminal is new, create a new record with a unique identifier
- The Criminal data inserted is with respect to the Police station where the criminal is arrested, there is data is stored

5.2.2 Efficiency Analysis :

- The time complexity of inserting into a CSV file is O(1).
- Space Complexity of CSV files is S(n).

5.3 : Creation of a Graph Using Linked List and Weighing the Edges:

5.3.1 Workflow :

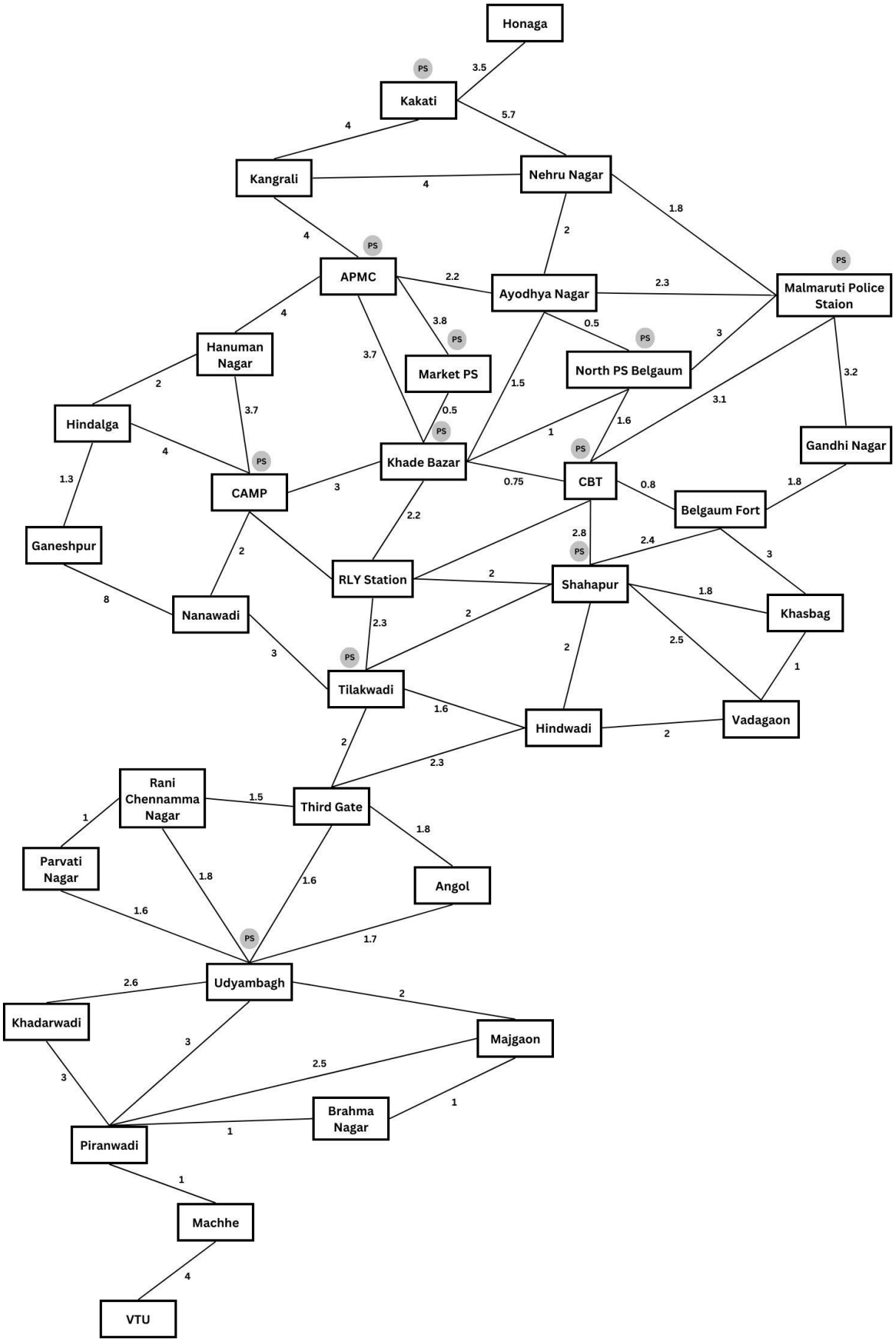
Created a graph using a linked list to represent the connectivity within Belgaum city, along with assigning weights to the edges.

- Gathered data from Google Maps containing information about locations and their connectivity in Belgaum city
- Identified significant locations in Belgaum city (nodes). These could include landmarks or intersections
- Create a linked list for each node to represent the edges (connections) from that node to other adjacent nodes
- For each node, populated the linked list with information about connected nodes (neighbours) and the associated edge weights. The edge weights represent distance between them
- Assigned accurate weights to the edges
- Validated the graph to ensure accuracy of the nodes which are marked as Police Stations

5.3.2 Efficiency Analysis :

- The time complexity of constructing the graph involves iterating through the data and adding nodes and edges. Assuming n is the number of locations in Belgaum, and m is the number of edges, the time complexity is O(n + m).
- The space complexity of the graph representation using a linked list involves storing information about nodes, edges, and their weights. Assuming each node and edge information requires constant space, the space complexity is O(n + m).

5.3.3 Created Graph :



5.4 : Criminal Search Across All Police Station Files Using KMP String Pattern Matching Algorithm

5.4.1 Workflow :

Searching for criminal records across all police station files using the Knuth-Morris-Pratt (KMP) string pattern matching algorithm involves breaking down the task into a series of steps.

- Received the criminal's name and age for identifying information as the input query
- Converted it to a format suitable for pattern matching
- Ignored the case of the pattern for better searching
- Initialized the KMP algorithm by preprocessing the query to construct the prefix function (LPS table)
- For each file, applied the KMP algorithm to search for occurrences of the query within the criminal records
- If matched then displayed all the corresponding field of particular criminal and corresponding Police Station

5.4.2 Efficiency Analysis :

- The time complexity of the KMP algorithm is O(m + n), where n is the length of text string, and m is the length of the pattern being searched.
- The space complexity of the KMP algorithm is O(m), where m is the length of the pattern.

5.5 : Menu System with Police Authentication with hidden Password

5.5.1 Workflow :

- User selects either "PUBLIC" or "POLICE" options
- Options include displaying city information, police stations, finding the nearest police station, searching for criminals, entering criminal data, and more
- If the user selects the "POLICE" option, they must enter a password for authentication
- The password entered during police authentication is **masked** for security
- The program loops, allowing users to navigate menus until they choose to exit

5.4.2 Efficiency Analysis :

- The time complexity for password authentication takes constant time O(1) as it involves only string comparison
- The space complexity is O(1) for storing the password

6. Conclusion

- We learned how to work as a team for better performance
- Consistency should be kept while working on a project
- Effective communication within our team
- Throughout the project, we navigated the intricacies of algorithmic design, data structures implementation, and coding challenges together
- Regular team meetings and discussions enabled us to share ideas, address challenges collectively, and ensure that everyone was on the same page regarding project goals and milestones
- As we encountered roadblocks, we learned to pivot our strategies, iterate on our solutions, and persevere through the iterative development process
- These soft skills, combined with the technical expertise gained, will undoubtedly shape my future experiences and contribute to my growth as a well-rounded team player in the field of computer science and beyond

7. References

[1] https://maps.google.com/ for creating a graph of Belgaum City
[2] https://www.geeksforgeeks.org/ for KMP string pattern matching algorithm and for the operations on CSV files

~*~*~*~*~*~*~*~