CSC 1012

Introduction to Computer Programming

Assignment (Individual)

Logistics Management System

Name: H. G. H. Kaveena

Student ID: AS20240503

Date: 24. 10. 2025

-------------TABLE OF CONTENTS-------------

1. Introduction
2. Objectives
3. System Overview
4. Functional Requirements
5. Sample Outputs
6. Challenges Faced
7. Conclusion
8. GitHub Repository Link
9. **Introduction**

This project, titled *“Logistics Management System,”* has been developed as part of the CSC 1012 – Introduction to Computer Programming course.  
The main aim of this system is to simulate a simple logistics and delivery management platform using the C Programming language. This system handles up to 30 cities and 50 delivery records. It helps manage cities, delivery routes, vehicles, and delivery requests while estimating time, cost, and fuel usage efficiently.

1. **Objectives**

This project aims to design and implement a menu-driven logistics management system in C that demonstrates practical programming skills and real-world application development.

The primary objectives of this project are:

* **City Management:** Enable users to add, remove, and rename cities in the logistics network
* **Distance Management:** Maintain a distance matrix between all city pairs
* **Cost Calculation:** Accurately calculate delivery costs based on multiple factors including distance, weight, and vehicle type
* **Performance Tracking:** Generate comprehensive reports on delivery operations
* **Data Persistence:** Save and load data

1. **Objectives**

This project aims to design and implement a menu-driven logistics management system in C that demonstrates practical programming skills and real-world application development.

The primary objectives of this project are:

* **City Management:** Enable users to add, remove, and rename cities in the logistics network
* **Distance Management:** Maintain a distance matrix between all city pairs
* **Cost Calculation:** Accurately calculate delivery costs based on multiple factors including distance, weight, and vehicle type
* **Performance Tracking:** Generate comprehensive reports on delivery operations
* **Data Persistence:** Save and load data
* Rename cities while preserving distance data
* Display all cities in numbered list format
* **Distance Management**
* Input distances between any two cities
* Automatic symmetry (distance A→B equals B→A)
* Prevention of self-loops (city to itself must be 0)
* Formatted distance table display
* **The system supports three vehicle types:**

Van, Truck, Lorry

* **Performance Tracking**
* Total number of deliveries completed
* Cumulative distance covered
* Average delivery time
* Total revenue generated
* Total profit earned
* Shortest and longest routes

1. **Functional Requirements**

1. City Management

* Add, rename, and remove cities.
* Display a distance matrix between all cities.

2. Vehicle Management

* Maintain data for three vehicle types (Van, Truck, Lorry).
* Store rate per km, capacity, and speed for each type.

3. Delivery Handling

* Enter delivery requests including source, destination, weight, and vehicle type.
* Validate weight and ensure source ≠ destination.

4. Calculations

* Delivery Cost = Distance × Rate × (1 + Weight / 10000)
* Time = Distance / Speed
* Fuel Used = Distance / Efficiency
* Total Cost = Delivery Cost + Fuel Cost
* Profit = 25% of Delivery Cost

5. Reports

* Total deliveries completed
* Total distance covered
* Average delivery time
* Total profit and revenue

1. **Sample Outputs**

**❖ Screenshots of Program Execution**

Below are the screenshots of program output for each function:

• Figure 1: Output of Main Menu System.

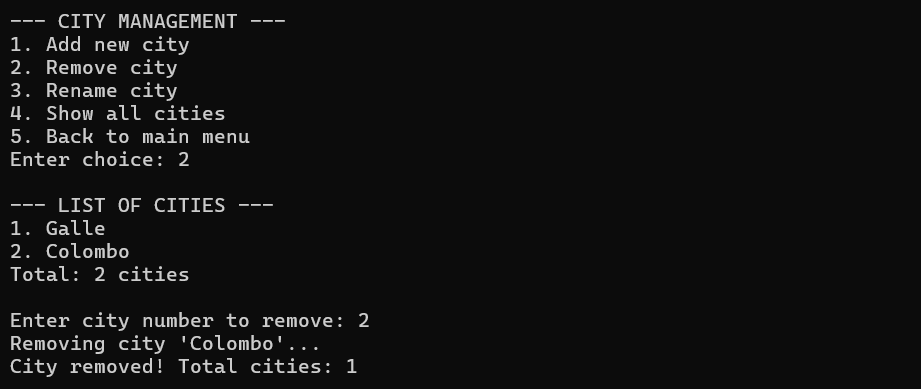


• Figure 2: Output of City Management

* 1. Output of Add new city function



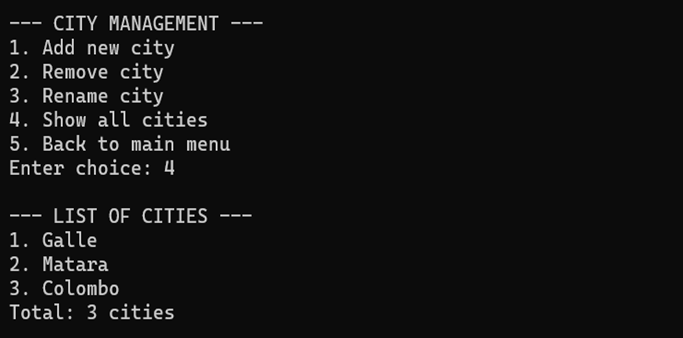
* 1. Output of Remove city functon



* 1. Output of Rename city function

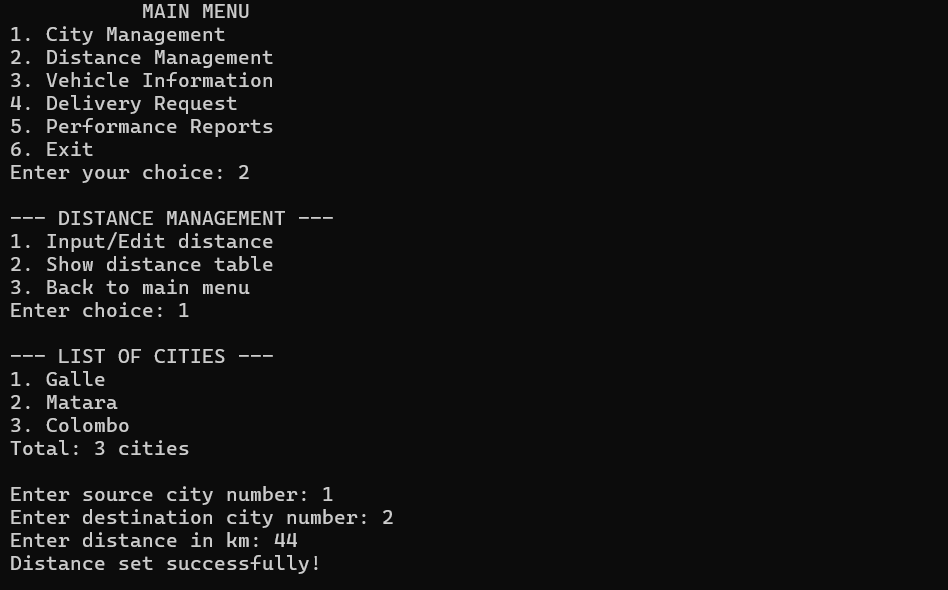


* 1. Output of Show all cities

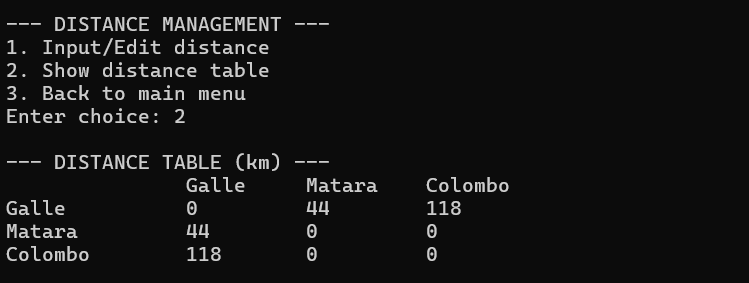


• Figure 3: Output of Distance Management

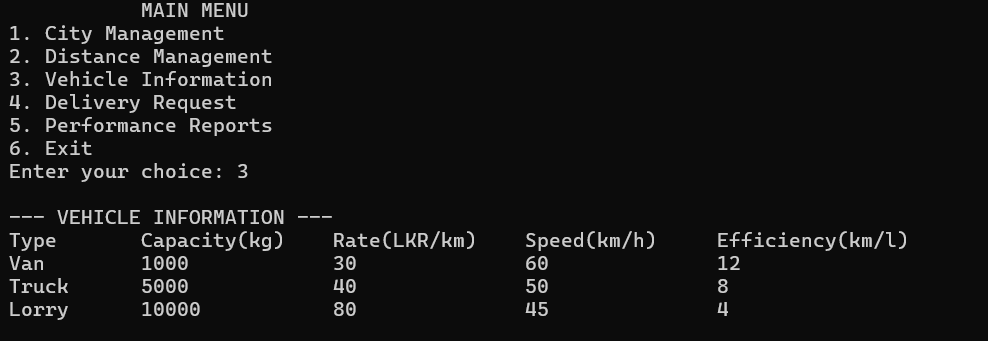
1. Input/edit distance



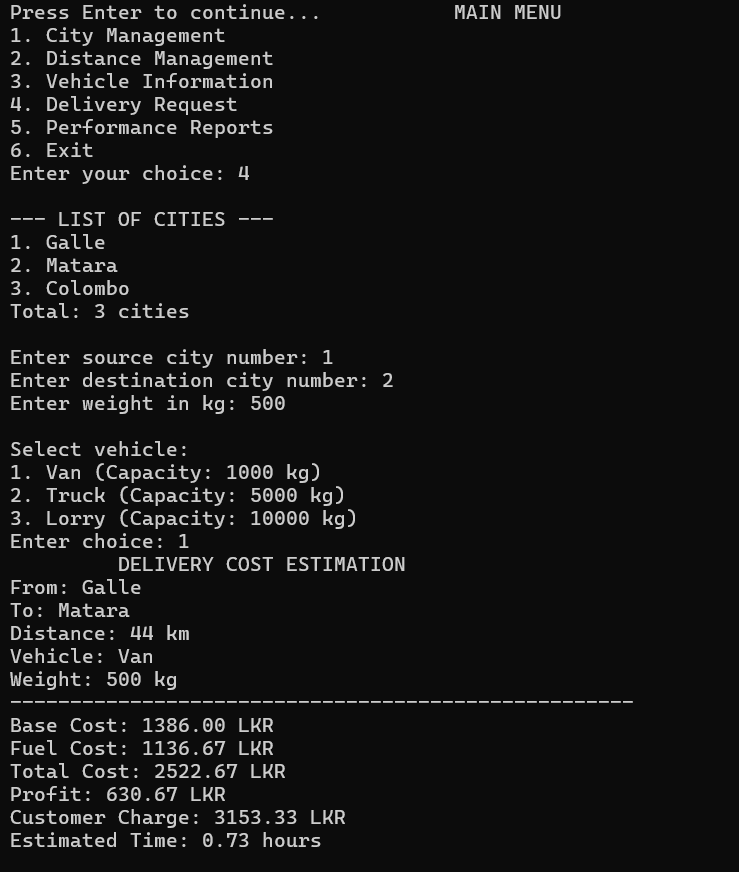
1. Show distance table



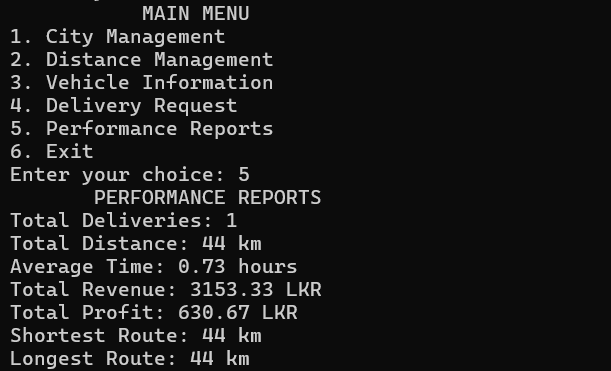
• Figure 4: Output of Vehicle information



• Figure 5: Output of Delivery Request



• Figure 6: Output of Performance Report



1. **Challenges Faced**

During the development process, the main challenges were:

* Handling **2D arrays** for city distances.
* Designing a clean **menu system** with functions.
* Ensuring **data validation** for delivery inputs.
* Implementing the **least-cost route algorithm**.

1. **Conclusion**

This project successfully demonstrates how to develop a simple yet effective logistics management system using C.

Through this assignment, I learned the importance of using modular functions, loops, and arrays efficiently.

It also improved my ability to write clean and structured code while solving real-world problems.

1. **GitHub Repository Link**

[https://github.com/HGHKAVEENA/logistics-management-system](https://github.com/HGHKAVEENA/logistics-management-system.git)