

Logistics Management System

Name: S D M N Perera

Std No: AS20240428

Course: CSC 1012 – Introduction to Programming

Repository Link : https://github.com/mudithperera33/Logistic_Management_System.git

1. Introduction

This project is a small logistics management system made using C language.

The aim was to make a simple program where user can add cities, manage distances, vehicles, deliveries and view reports.

It is a console program with menus. Each part works from a separate function so it is easy to understand.

2. What the Program Does

The program has 5 main parts:

1. **City Management** – Add, list, remove and rename cities.
2. **Distance Management** – Add or view distances between cities.
3. **Vehicle Management** – Show van, truck and lorry details.
4. **Delivery Management** – Add deliveries with city, vehicle and weight.
5. **Reports** – Show total and average cost of deliveries.

The program also saves data to text files so next time it can load again without losing cities or distances.

3. How It Works

When the program starts, it loads data from two files called cities.txt and distance.txt.

Then the main menu appears with the 6 options.

Each menu option calls a different function like cityMenu(), distanceMenu() or deliveryMenu().

Example:

- addCity() lets the user enter a city name and adds it to the list.
- editDistance() lets the user enter distance between two cities.
- addDelivery() lets the user pick cities and vehicle type.

Then the program calculates cost, time and fuel like this:

```
cost = rate * distance
time = distance / speed
fuel = distance / efficiency
```

At the end when exiting, all data is saved again into the text files.

4. Problems I Faced

At first, I had many problems with arrays and functions. When I used normal variables for city count, it did not update correctly in other functions. Then I learned I had to use pointer (`int *cityCount`) to make it work.

File handling was hard too. When reading cities from file, there was an extra newline character. I did not know why the city name looked wrong in the program. Then I learned to use

```
strcspn(cities[i], "\n")
```

to remove the newline.

I had problems with **distance table**. At first, distances were not saving correctly. I fixed it by checking loops and writing distances row by row. Also, I had to make sure distance from city to itself is 0.

Delivery management was also difficult. Connecting deliveries with correct cities and distances took some tries. At first, report was giving wrong total cost because I used wrong index.

5. Testing

I tested by adding few cities like Colombo, Galle, Kandy.

Then I added distances and checked the table.

After that I added some deliveries using different vehicles and weights.

The cost, time and fuel values looked correct.

Then I closed and opened program again to check file saving — it worked fine.

6. Limitations

- Program does not check for wrong inputs (like typing letters instead of numbers).
- It can't edit or delete deliveries.
- I didn't use struct or dynamic memory, only arrays.
- Report only shows total and average cost.

Maybe in future I can improve it by adding structures and more options.

7. Conclusion

This project helped me understand functions, arrays and file handling better.

I learned how to break big problem into small parts and use menus to make it easy.

Even though there were small bugs and some things not perfect, I am happy that the program works properly in the end.