Traffic Fine Management System

Project Title: Traffic Fine Management System

Student Name: Aravind VS

Register Number: 921724102020

Department: CSE

College Name: SIT

Course: Programming in C

Guide / Trainer Name: PUGAZHAL MARAN

Submission Date: 24/02/2026

Abstract:

The Traffic Fine Management System is designed to manage traffic violation records efficiently using C programming. It allows users to add, display, search, update, and delete records with file handling support.

Introduction:

Traffic rule violations are common, and managing fine records manually can be difficult and time-consuming. This project provides a simple digital solution to store and manage traffic violation data.

The project is useful because:

- It reduces manual record-keeping errors.
- It allows quick searching and updating of records.
- It stores data permanently using files.

This system can be used in:

- Traffic police departments
- Transport offices
- Educational learning environments

Objectives:

- Understand structured programming

- Use loops and conditions

- Work with structures and file handling

- Improve logical thinking

Tools & Technology:

Programming Language: C

Compiler: GCC / CodeBlocks / Turbo C

Platform: Windows / Linux

System Requirements:

Hardware: Computer with minimum 4GB RAM

Software: C Compiler and IDE

Methodology / Algorithm:

1. Start program

2. Display menu

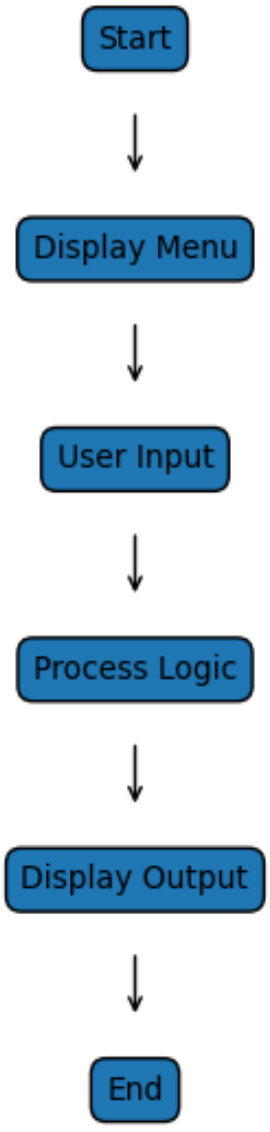
3. Read user input

4. Process logic

5. Display output

6. End program

Flowchart:



Program:

```
#include <stdio.h>

#include <stdlib.h>
#include <string.h>

struct fine {
   char vehicleNo[20];
   char ownerName[50];
   char violation[50];
   int fineAmount;
   char paymentStatus[20];
};

void addRecord();
void displayRecords();
void searchRecord();
void updatePayment();
void deleteRecord();

int main() {
   int choice;

   while (1) {
      printf("\n===== Traffic Fine Management System =====\n");
      printf("1. Add Violation Record\n");
      printf("2. Display All Records\n");
      printf("3. Search by Vehicle Number\n");
      printf("4. Update Payment Status\n");
      printf("5. Delete Record\n");
      printf("6. Exit\n");
      printf("Enter your choice: ");
      scanf("%d", &choice);

      switch (choice) {
         case 1: addRecord(); break;
         case 2: displayRecords(); break;
         case 3: searchRecord(); break;
         case 4: updatePayment(); break;
         case 5: deleteRecord(); break;
         case 6: exit(0);
         default: printf("Invalid choice!\n");
      }
   }
}
```

```
int calculateFine(int type) {
    switch (type) {
        case 1: return 500; // No Helmet
        case 2: return 1000; // Signal Jump
        case 3: return 1500; // Drunk & Drive
        case 4: return 800; // No License
        default: return 0;
    }
}

void addRecord() {
    FILE *fp = fopen("fine.dat", "ab");
    struct fine f;
    int type;

    printf("Enter Vehicle Number: ");
    scanf("%s", f.vehicleNo);
    printf("Enter Owner Name: ");
    scanf("%s", f.ownerName);

    printf("\nViolation Types:\n");
    printf("1. No Helmet (500)\n");
    printf("2. Signal Jump (1000)\n");
    printf("3. Drunk & Drive (1500)\n");
    printf("4. No License (800)\n");
    printf("Select violation type: ");
    scanf("%d", &type);

    f.fineAmount = calculateFine(type);

    switch(type) {
        case 1: strcpy(f.violation, "No Helmet"); break;
        case 2: strcpy(f.violation, "Signal Jump"); break;
        case 3: strcpy(f.violation, "Drunk & Drive"); break;
        case 4: strcpy(f.violation, "No License"); break;
        default: strcpy(f.violation, "Unknown");
    }

    strcpy(f.paymentStatus, "Pending");

    fwrite(&f, sizeof(f), 1, fp);
    fclose(fp);

    printf("Record Added Successfully!\n");
}
```

```
void displayRecords() {
    FILE *fp = fopen("fine.dat", "rb");
    struct fine f;

    if (fp == NULL) {
        printf("No records found!\n");
        return;
    }

    printf("\n--- Traffic Fine Records ---\n");
    while (fread(&f, sizeof(f), 1, fp)) {
        printf("Vehicle: %s | Owner: %s | Violation: %s | Fine: %d | Status: %s\n",
               f.vehicleNo, f.ownerName, f.violation, f.fineAmount, f.paymentStatus);
    }

    fclose(fp);
}

void searchRecord() {
    FILE *fp = fopen("fine.dat", "rb");
    struct fine f;
    char vehicleNo[20];
    int found = 0;

    printf("Enter Vehicle Number to Search: ");
    scanf("%s", vehicleNo);

    while (fread(&f, sizeof(f), 1, fp)) {
        if (strcmp(f.vehicleNo, vehicleNo) == 0) {
            printf("Record Found!\n");
            printf("Owner: %s | Violation: %s | Fine: %d | Status: %s\n",
                   f.ownerName, f.violation, f.fineAmount, f.paymentStatus);
            found = 1;
            break;
        }
    }

    if (!found)
        printf("Record Not Found!\n");

    fclose(fp);
}

void updatePayment() {
    FILE *fp = fopen("fine.dat", "rb+");
    struct fine f;
```

```
    char vehicleNo[20];
    int found = 0;

    printf("Enter Vehicle Number to Update Payment: ");
    scanf("%s", vehicleNo);

    while (fread(&f, sizeof(f), 1, fp)) {
        if (strcmp(f.vehicleNo, vehicleNo) == 0) {
            strcpy(f.paymentStatus, "Paid");
            fseek(fp, -sizeof(f), SEEK_CUR);
            fwrite(&f, sizeof(f), 1, fp);
            printf("Payment Updated Successfully!\n");
            found = 1;
            break;
        }
    }

    if (!found)
        printf("Record Not Found!\n");

    fclose(fp);
}

void deleteRecord() {
    FILE *fp = fopen("fine.dat", "rb");
    FILE *temp = fopen("temp.dat", "wb");
    struct fine f;
    char vehicleNo[20];
    int found = 0;

    printf("Enter Vehicle Number to Delete: ");
    scanf("%s", vehicleNo);

    while (fread(&f, sizeof(f), 1, fp)) {
        if (strcmp(f.vehicleNo, vehicleNo) != 0) {
            fwrite(&f, sizeof(f), 1, temp);
        } else {
            found = 1;
        }
    }

    fclose(fp);
    fclose(temp);

    remove("fine.dat");
    rename("temp.dat", "fine.dat");
```

```
    if (found)
        printf("Record Deleted Successfully!\n");
    else
        printf("Record Not Found!\n");
}
```

Output:

```
===== Traffic Fine Management System =====

1. Add Violation Record
2. Display All Records
3. Search by Vehicle Number
4. Update Payment Status
5. Delete Record
6. Exit
Enter your choice: 1

Enter Vehicle Number: TN58AB1234
Enter Owner Name: Aravind

Violation Types:
1. No Helmet (500)
2. Signal Jump (1000)
3. Drunk & Drive (1500)
4. No License (800)
Select violation type: 2

Record Added Successfully!
```

Result:

The program executed successfully and produced the expected output.

Conclusion: The project demonstrates basic C programming concepts with file handling.

Future Enhancements: Add GUI, authentication, and database integration.

References: C Programming books, Online tutorials, Class notes.