

```

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

#define MAX_SLOTS 10

typedef struct {
    char license_plate[15];
    int slot_number;
    int is_occupied;
} ParkingSlot;

ParkingSlot parking_slots[MAX_SLOTS];

void initialize_slots() {
    for (int i = 0; i < MAX_SLOTS; i++) {
        parking_slots[i].slot_number = i + 1;
        parking_slots[i].is_occupied = 0;
        strcpy(parking_slots[i].license_plate, "");
    }
}

void display_menu() {
    printf("\nParking Lot Management System\n");
    printf("1. Park Vehicle\n");
    printf("2. Remove Vehicle\n");
    printf("3. Display Available Slots\n");
    printf("4. Exit\n");
    printf("Enter your choice: ");
}

void park_vehicle() {
    int slot_number;
    char license_plate[15];

    printf("Enter slot number to park (1-%d): ", MAX_SLOTS);
    scanf("%d", &slot_number);

    if (slot_number < 1 || slot_number > MAX_SLOTS) {
        printf("Invalid slot number.\n");
        return;
    }

    if (parking_slots[slot_number - 1].is_occupied) {
        printf("Slot is already occupied.\n");
        return;
    }

    printf("Enter vehicle license plate: ");
    scanf("%s", license_plate);

    parking_slots[slot_number - 1].is_occupied = 1;
    strcpy(parking_slots[slot_number - 1].license_plate, license_plate);

    printf("Vehicle parked successfully in slot %d.\n", slot_number);
}

```

```

void remove_vehicle() {
    int slot_number;

    printf("Enter slot number to remove vehicle (1-%d): ", MAX_SLOTS);
    scanf("%d", &slot_number);

    if (slot_number < 1 || slot_number > MAX_SLOTS) {
        printf("Invalid slot number.\n");
        return;
    }
    if (!parking_slots[slot_number - 1].is_occupied) {
        printf("Slot is already empty.\n");
        return;
    }
    parking_slots[slot_number - 1].is_occupied = 0;
    strcpy(parking_slots[slot_number - 1].license_plate, "");

    printf("Vehicle removed from slot %d successfully.\n", slot_number);
}

void display_available_slots() {
    printf("\nAvailable Parking Slots:\n");
    for (int i = 0; i < MAX_SLOTS; i++) {
        if (!parking_slots[i].is_occupied) {
            printf("Slot %d: Available\n", i + 1);
        }
    }
}

int main() {
    int choice;
    initialize_slots();

    do {
        display_menu();
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                park_vehicle();
                break;
            case 2:
                remove_vehicle();
                break;
            case 3:
                display_available_slots();
                break;
            case 4:
                printf("Exiting program.\n");
                break;
            default:
                printf("Invalid choice. Please try again.\n");
        }
    } while (choice != 4);
    return 0;
}

```