Requirements

1. Define Data Types

Appliance Data Structure:

Define a structure Appliance to represent each smart appliance in the home.

Fields:

applianceID (integer): Unique ID for the appliance.

applianceName (string): Name of the appliance (e.g., "Light", "Fan", "Thermostat").

isOn (integer): 1 if the appliance is ON, 0 if OFF.

powerConsumption (float): Power consumption of the appliance in watts.

Union for Appliance Settings:

Define a union ApplianceSettings to represent optional configurations for the appliance, such as:

temperature (float): Temperature setting (for AC or thermostat).

brightness (float): Brightness level (for lights).

fanSpeed (integer): Speed setting (for fans).

Home Data Structure:

Define a structure HomeAutomation to store:

numAppliances (integer): Total number of appliances connected.

An array of Appliance structures.

An array of ApplianceSettings unions for each appliance.

2. Features to Implement

Dynamic Memory Allocation:

Dynamically allocate memory for an array of Appliance structures and ApplianceSettings unions based on the number of appliances (N) in the home.

Input and Output:

Input the details of each appliance, including its name, power consumption, and optional settings (e.g., temperature, brightness).

Display the details of all appliances, including their settings.

Appliance Control:

Turn appliances ON or OFF by updating their isOn field.

Update specific settings (e.g., adjust temperature, brightness, or fan speed).

Power Management:

Calculate the total power consumption of all active appliances.

Identify the appliance consuming the most power.

Sorting and Analysis:

Sort appliances by power consumption in descending order.

List all appliances that are currently ON.

Typedef Usage:

Use typedef to simplify the code for Appliance and ApplianceSettings.

Example Program Flow

Menu-Driven Interface:

Provide a user-friendly menu with options:

Input Appliance Data

Display All Appliances

Control Appliances (Turn ON/OFF)

Update Appliance Settings

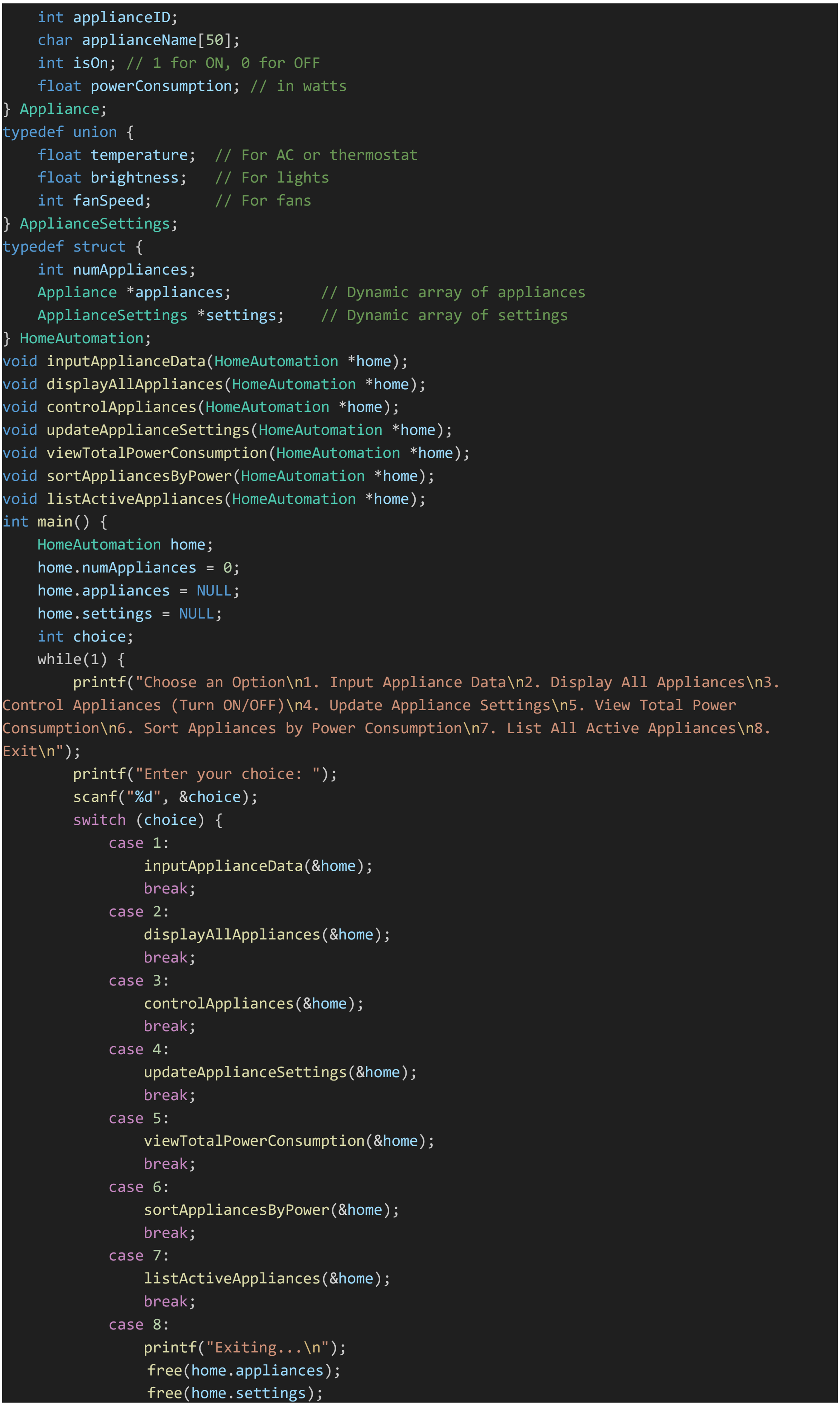
View Total Power Consumption

Sort Appliances by Power Consumption

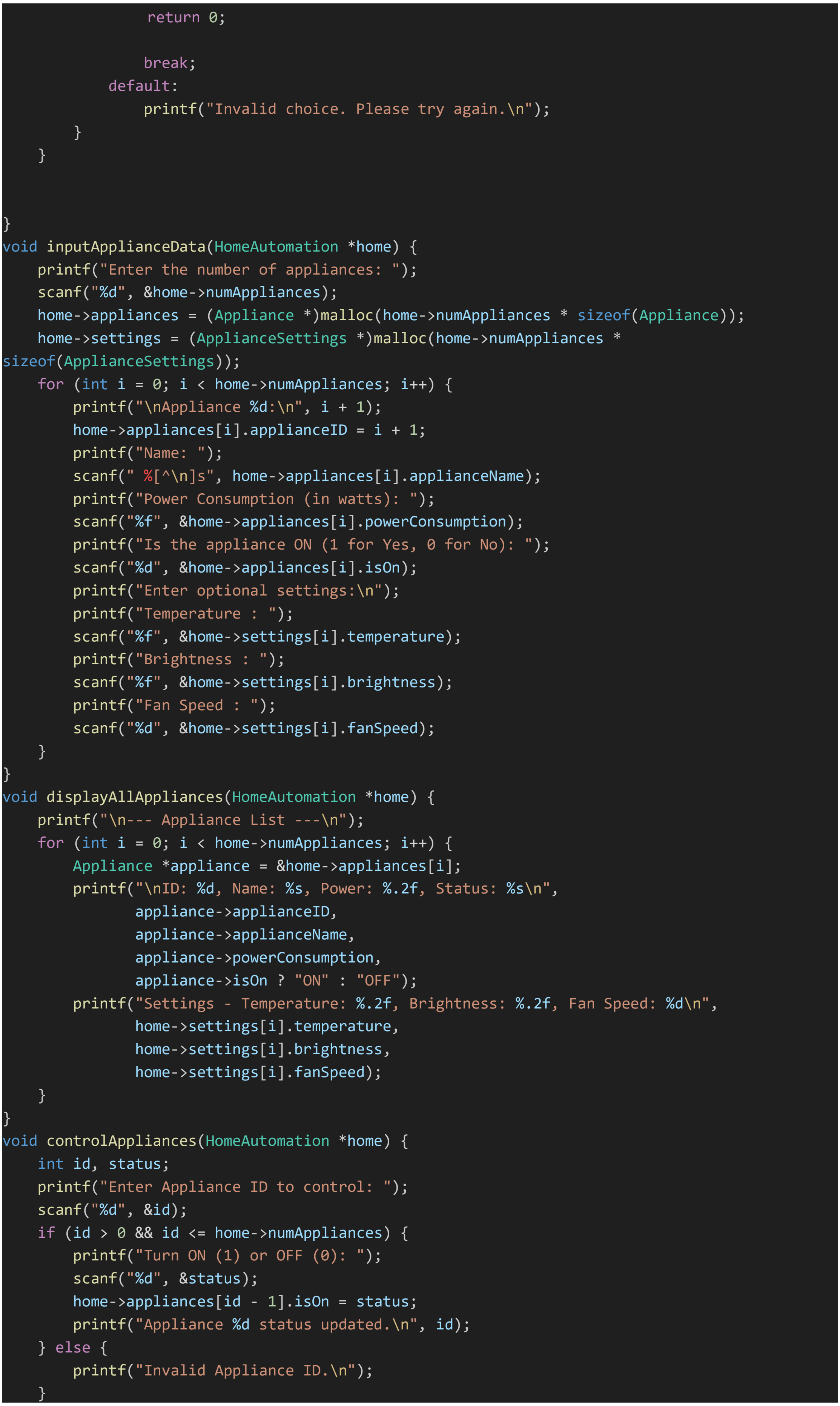
Exit

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


typedef struct {
```



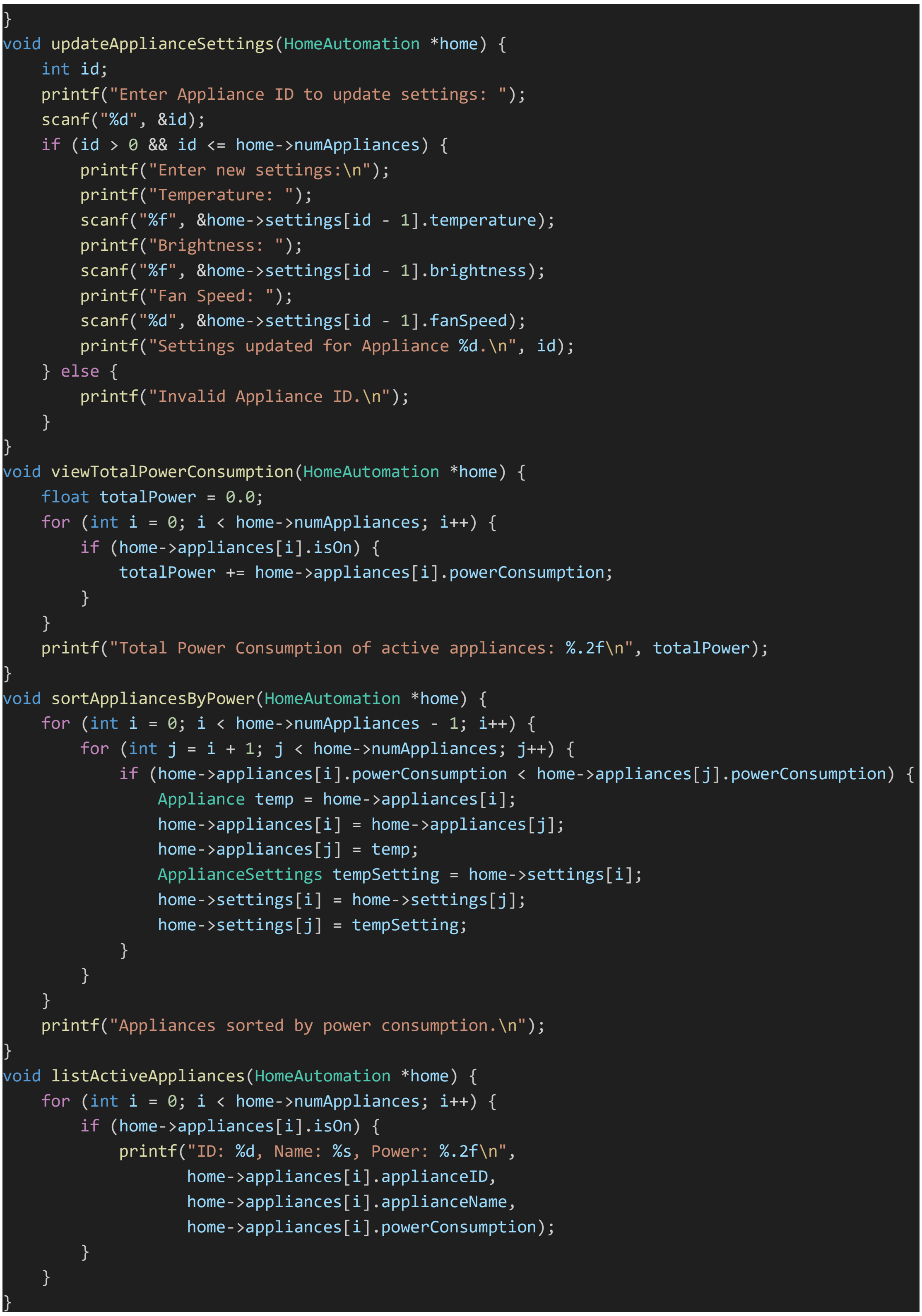
```
    int applianceID;
    char applianceName[50];
    int isOn; // 1 for ON, 0 for OFF
    float powerConsumption; // in watts
} Appliance;
typedef union {
    float temperature;  // For AC or thermostat
    float brightness;   // For lights
    int fanSpeed;       // For fans
} ApplianceSettings;
typedef struct {
    int numAppliances;
    Appliance *appliances;          // Dynamic array of appliances
    ApplianceSettings *settings;    // Dynamic array of settings
} HomeAutomation;
void inputApplianceData(HomeAutomation *home);
void displayAllAppliances(HomeAutomation *home);
void controlAppliances(HomeAutomation *home);
void updateApplianceSettings(HomeAutomation *home);
void viewTotalPowerConsumption(HomeAutomation *home);
void sortAppliancesByPower(HomeAutomation *home);
void listActiveAppliances(HomeAutomation *home);
int main() {
    HomeAutomation home;
    home.numAppliances = 0;
    home.appliances = NULL;
    home.settings = NULL;
    int choice;
    while(1) {
        printf("Choose an Option\n1. Input Appliance Data\n2. Display All Appliances\n3.
Control Appliances (Turn ON/OFF)\n4. Update Appliance Settings\n5. View Total Power
Consumption\n6. Sort Appliances by Power Consumption\n7. List All Active Appliances\n8.
Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);
        switch (choice) {
            case 1:
                inputApplianceData(&home);
                break;
            case 2:
                displayAllAppliances(&home);
                break;
            case 3:
                controlAppliances(&home);
                break;
            case 4:
                updateApplianceSettings(&home);
                break;
            case 5:
                viewTotalPowerConsumption(&home);
                break;
            case 6:
                sortAppliancesByPower(&home);
                break;
            case 7:
                listActiveAppliances(&home);
                break;
            case 8:
                printf("Exiting...\n");
                 free(home.appliances);
                 free(home.settings);
```



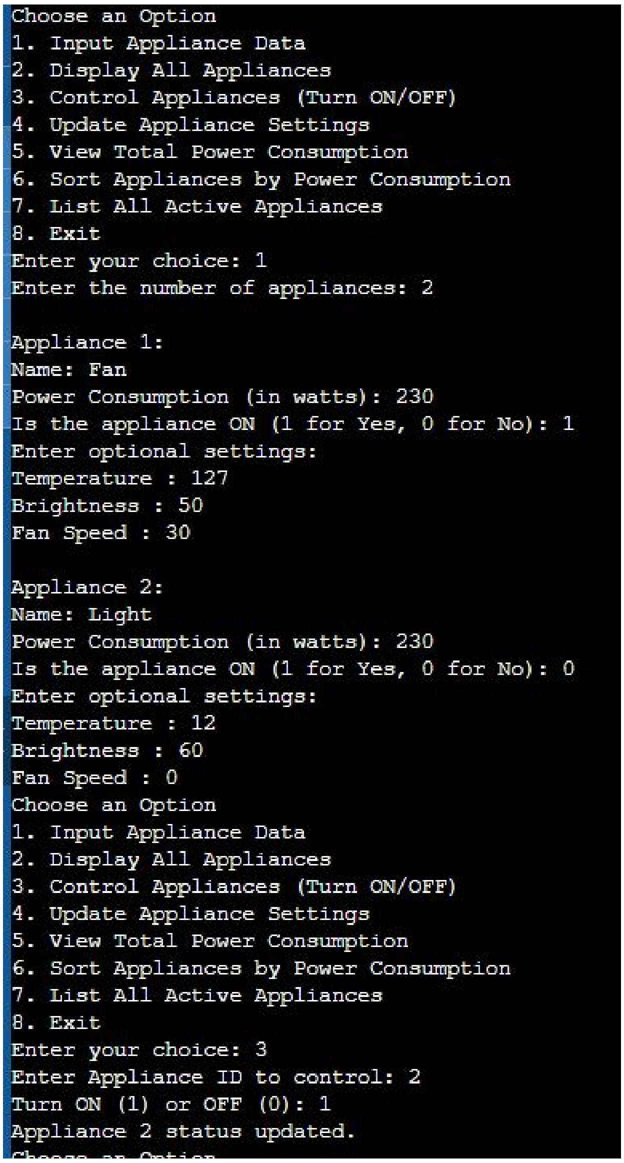
```c
                return 0;

                break;
            default:
                printf("Invalid choice. Please try again.\n");
        }
    }


}
void inputApplianceData(HomeAutomation *home) {
    printf("Enter the number of appliances: ");
    scanf("%d", &home->numAppliances);
    home->appliances = (Appliance *)malloc(home->numAppliances * sizeof(Appliance));
    home->settings = (ApplianceSettings *)malloc(home->numAppliances *
sizeof(ApplianceSettings));
    for (int i = 0; i < home->numAppliances; i++) {
        printf("\nAppliance %d:\n", i + 1);
        home->appliances[i].applianceID = i + 1;
        printf("Name: ");
        scanf(" %[^\n]s", home->appliances[i].applianceName);
        printf("Power Consumption (in watts): ");
        scanf("%f", &home->appliances[i].powerConsumption);
        printf("Is the appliance ON (1 for Yes, 0 for No): ");
        scanf("%d", &home->appliances[i].isOn);
        printf("Enter optional settings:\n");
        printf("Temperature : ");
        scanf("%f", &home->settings[i].temperature);
        printf("Brightness : ");
        scanf("%f", &home->settings[i].brightness);
        printf("Fan Speed : ");
        scanf("%d", &home->settings[i].fanSpeed);
    }
}
void displayAllAppliances(HomeAutomation *home) {
    printf("\n--- Appliance List ---\n");
    for (int i = 0; i < home->numAppliances; i++) {
        Appliance *appliance = &home->appliances[i];
        printf("\nID: %d, Name: %s, Power: %.2f, Status: %s\n",
               appliance->applianceID,
               appliance->applianceName,
               appliance->powerConsumption,
               appliance->isOn ? "ON" : "OFF");
        printf("Settings - Temperature: %.2f, Brightness: %.2f, Fan Speed: %d\n",
               home->settings[i].temperature,
               home->settings[i].brightness,
               home->settings[i].fanSpeed);
    }
}
void controlAppliances(HomeAutomation *home) {
    int id, status;
    printf("Enter Appliance ID to control: ");
    scanf("%d", &id);
    if (id > 0 && id <= home->numAppliances) {
        printf("Turn ON (1) or OFF (0): ");
        scanf("%d", &status);
        home->appliances[id - 1].isOn = status;
        printf("Appliance %d status updated.\n", id);
    } else {
        printf("Invalid Appliance ID.\n");
    }
```



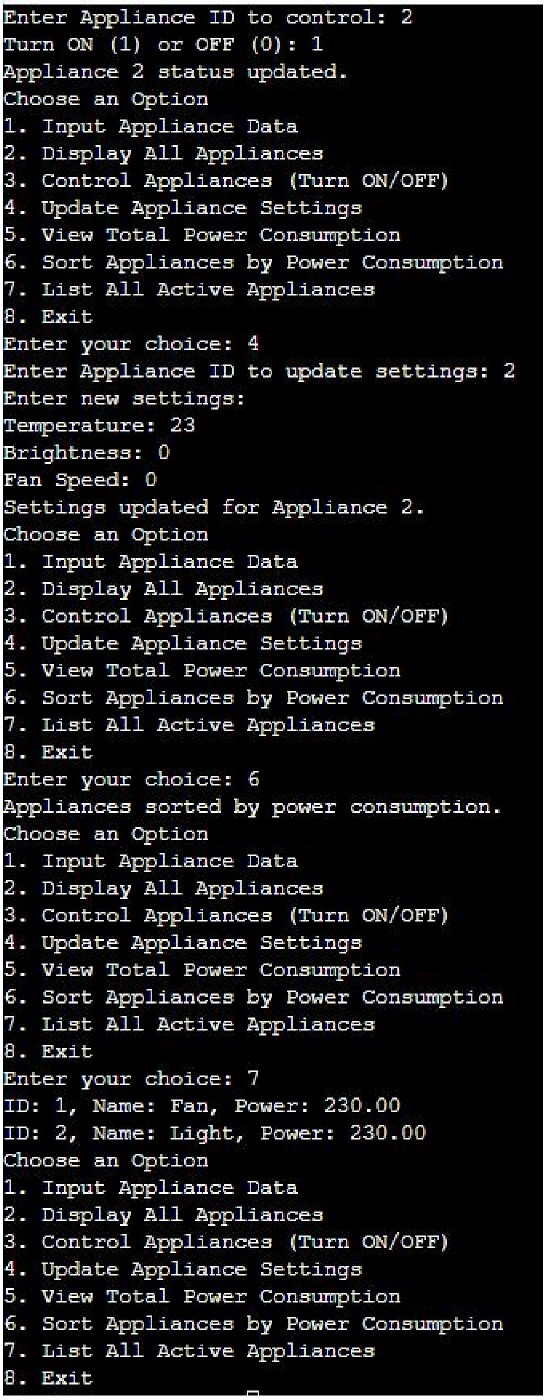
```c
}
void updateApplianceSettings(HomeAutomation *home) {
    int id;
    printf("Enter Appliance ID to update settings: ");
    scanf("%d", &id);
    if (id > 0 && id <= home->numAppliances) {
        printf("Enter new settings:\n");
        printf("Temperature: ");
        scanf("%f", &home->settings[id - 1].temperature);
        printf("Brightness: ");
        scanf("%f", &home->settings[id - 1].brightness);
        printf("Fan Speed: ");
        scanf("%d", &home->settings[id - 1].fanSpeed);
        printf("Settings updated for Appliance %d.\n", id);
    } else {
        printf("Invalid Appliance ID.\n");
    }
}
void viewTotalPowerConsumption(HomeAutomation *home) {
    float totalPower = 0.0;
    for (int i = 0; i < home->numAppliances; i++) {
        if (home->appliances[i].isOn) {
            totalPower += home->appliances[i].powerConsumption;
        }
    }
    printf("Total Power Consumption of active appliances: %.2f\n", totalPower);
}
void sortAppliancesByPower(HomeAutomation *home) {
    for (int i = 0; i < home->numAppliances - 1; i++) {
        for (int j = i + 1; j < home->numAppliances; j++) {
            if (home->appliances[i].powerConsumption < home->appliances[j].powerConsumption) {
                Appliance temp = home->appliances[i];
                home->appliances[i] = home->appliances[j];
                home->appliances[j] = temp;
                ApplianceSettings tempSetting = home->settings[i];
                home->settings[i] = home->settings[j];
                home->settings[j] = tempSetting;
            }
        }
    }
    printf("Appliances sorted by power consumption.\n");
}
void listActiveAppliances(HomeAutomation *home) {
    for (int i = 0; i < home->numAppliances; i++) {
        if (home->appliances[i].isOn) {
            printf("ID: %d, Name: %s, Power: %.2f\n",
                   home->appliances[i].applianceID,
                   home->appliances[i].applianceName,
                   home->appliances[i].powerConsumption);
        }
    }
}
```



```
Choose an Option
1. Input Appliance Data
2. Display All Appliances
3. Control Appliances (Turn ON/OFF)
4. Update Appliance Settings
5. View Total Power Consumption
6. Sort Appliances by Power Consumption
7. List All Active Appliances
8. Exit
Enter your choice: 1
Enter the number of appliances: 2

Appliance 1:
Name: Fan
Power Consumption (in watts): 230
Is the appliance ON (1 for Yes, 0 for No): 1
Enter optional settings:
Temperature : 127
Brightness : 50
Fan Speed : 30

Appliance 2:
Name: Light
Power Consumption (in watts): 230
Is the appliance ON (1 for Yes, 0 for No): 0
Enter optional settings:
Temperature : 12
Brightness : 60
Fan Speed : 0
Choose an Option
1. Input Appliance Data
2. Display All Appliances
3. Control Appliances (Turn ON/OFF)
4. Update Appliance Settings
5. View Total Power Consumption
6. Sort Appliances by Power Consumption
7. List All Active Appliances
8. Exit
Enter your choice: 3
Enter Appliance ID to control: 2
Turn ON (1) or OFF (0): 1
Appliance 2 status updated.
Choose an Option
```



```
Enter Appliance ID to control: 2
Turn ON (1) or OFF (0): 1
Appliance 2 status updated.
Choose an Option
1. Input Appliance Data
2. Display All Appliances
3. Control Appliances (Turn ON/OFF)
4. Update Appliance Settings
5. View Total Power Consumption
6. Sort Appliances by Power Consumption
7. List All Active Appliances
8. Exit
Enter your choice: 4
Enter Appliance ID to update settings: 2
Enter new settings:
Temperature: 23
Brightness: 0
Fan Speed: 0
Settings updated for Appliance 2.
Choose an Option
1. Input Appliance Data
2. Display All Appliances
3. Control Appliances (Turn ON/OFF)
4. Update Appliance Settings
5. View Total Power Consumption
6. Sort Appliances by Power Consumption
7. List All Active Appliances
8. Exit
Enter your choice: 6
Appliances sorted by power consumption.
Choose an Option
1. Input Appliance Data
2. Display All Appliances
3. Control Appliances (Turn ON/OFF)
4. Update Appliance Settings
5. View Total Power Consumption
6. Sort Appliances by Power Consumption
7. List All Active Appliances
8. Exit
Enter your choice: 7
ID: 1, Name: Fan, Power: 230.00
ID: 2, Name: Light, Power: 230.00
Choose an Option
1. Input Appliance Data
2. Display All Appliances
3. Control Appliances (Turn ON/OFF)
4. Update Appliance Settings
5. View Total Power Consumption
6. Sort Appliances by Power Consumption
7. List All Active Appliances
8. Exit
```