

```

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

// Structures
struct Item {
    int id;
    char name[50];
    int quantity;
};

struct Membership {
    int id;
    char vendorName[50];
    int duration; // in months
};

struct Order {
    int id;
    int amount;
    char status[20];
};

// Global Variables
struct Item items[10];
struct Membership memberships[10];
struct Order orders[10];

int itemCount = 0, membershipCount = 0, orderCount = 0;

// Function Declarations
void login();
void adminMenu();
void vendorMenu();
void userMenu();
void manageMembership();
void addItem();
void deleteItem();
void viewItems();
void userPayment();
void userOrderStatus();
void vendorRequest();
void adminVendorManagement();
void adminUserManagement();

int main() {
    printf("Welcome to the Library Management System\n");
    login();
    return 0;
}

```

```

}

void login() {
    char username[20], password[20];

    printf("\nLogin Page\n");
    printf("Enter Username: ");
    scanf("%s", username);
    printf("Enter Password: ");
    scanf("%s", password);

    if (strcmp(username, "admin") == 0 && strcmp(password, "admin123") == 0) {
        printf("\nWelcome Admin!\n");
        adminMenu();
    } else if (strcmp(username, "vendor") == 0 && strcmp(password, "vendor123") == 0) {
        printf("\nWelcome Vendor!\n");
        vendorMenu();
    } else if (strcmp(username, "user") == 0 && strcmp(password, "user123") == 0) {
        printf("\nWelcome User!\n");
        userMenu();
    } else {
        printf("Invalid credentials. Try again.\n");
        login();
    }
}

```

// Admin Menu

```

void adminMenu() {
    int choice;
    do {
        printf("\n--- Admin Menu ---\n");
        printf("1. Manage Memberships\n");
        printf("2. Vendor Management\n");
        printf("3. User Management\n");
        printf("4. Logout\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1: manageMembership(); break;
            case 2: adminVendorManagement(); break;
            case 3: adminUserManagement(); break;
            case 4: printf("Logging out...\n"); login(); break;
            default: printf("Invalid choice. Try again.\n");
        }
    } while (choice != 4);
}

```

```

// Manage Membership
void manageMembership() {
    int duration, choice;
    printf("\n--- Add/Update Memberships ---\n");
    printf("Enter Vendor Name: ");
    scanf("%s", memberships[membershipCount].vendorName);
    printf("Select Membership Duration (6, 12, or 24 months): ");
    scanf("%d", &duration);
    memberships[membershipCount].duration = duration;
    memberships[membershipCount].id = membershipCount + 1;
    membershipCount++;
    printf("Membership added successfully.\n");
}

// Vendor Menu
void vendorMenu() {
    int choice;
    do {
        printf("\n--- Vendor Menu ---\n");
        printf("1. Add Item\n");
        printf("2. Delete Item\n");
        printf("3. View Items\n");
        printf("4. User Requests\n");
        printf("5. Logout\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1: addItem(); break;
            case 2: deleteItem(); break;
            case 3: viewItems(); break;
            case 4: vendorRequest(); break;
            case 5: printf("Logging out...\n"); login(); break;
            default: printf("Invalid choice. Try again.\n");
        }
    } while (choice != 5);
}

void addItem() {
    printf("\n--- Add New Item ---\n");
    printf("Enter Item Name: ");
    scanf("%s", items[itemCount].name);
    printf("Enter Quantity: ");
    scanf("%d", &items[itemCount].quantity);
    items[itemCount].id = itemCount + 1;
    printf("Item added successfully. ID: %d\n", items[itemCount].id);
    itemCount++;
}

```

```

void deleteItem() {
    int id;
    printf("\n--- Delete Item ---\n");
    printf("Enter Item ID to delete: ");
    scanf("%d", &id);
    if (id > 0 && id <= itemCount) {
        printf("Item ID %d deleted.\n", id);
        items[id - 1].quantity = 0; // Mark as deleted
    } else {
        printf("Invalid Item ID.\n");
    }
}

void viewItems() {
    printf("\n--- View Items ---\n");
    printf("ID\tName\tQuantity\n");
    for (int i = 0; i < itemCount; i++) {
        if (items[i].quantity > 0) {
            printf("%d\t%s\t%d\n", items[i].id, items[i].name, items[i].quantity);
        }
    }
}

void vendorRequest() {
    printf("\n--- User Requests ---\n");
    printf("No requests at the moment.\n");
}

// User Menu
void userMenu() {
    int choice;
    do {
        printf("\n--- User Menu ---\n");
        printf("1. View Vendors\n");
        printf("2. Cart and Payment\n");
        printf("3. Order Status\n");
        printf("4. Logout\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1: viewItems(); break;
            case 2: userPayment(); break;
            case 3: userOrderStatus(); break;
            case 4: printf("Logging out...\n"); login(); break;
            default: printf("Invalid choice. Try again.\n");
        }
    }
}

```

```
    } while (choice != 4);  
}
```

```
void userPayment() {  
    int amount;  
    printf("\n--- Payment ---\n");  
    printf("Enter Payment Amount: ");  
    scanf("%d", &amount);  
    orders[orderCount].id = orderCount + 1;  
    orders[orderCount].amount = amount;  
    strcpy(orders[orderCount].status, "Pending");  
    printf("Payment Successful! Order ID: %d, Status: %s\n", orders[orderCount].id,  
orders[orderCount].status);  
    orderCount++;  
}
```

```
void userOrderStatus() {  
    printf("\n--- Order Status ---\n");  
    for (int i = 0; i < orderCount; i++) {  
        printf("Order ID: %d, Amount: %d, Status: %s\n", orders[i].id, orders[i].amount,  
orders[i].status);  
    }  
}
```

```
void adminVendorManagement() {  
    printf("\n--- Vendor Management ---\n");  
    printf("Managing vendors...\n");  
}
```

```
void adminUserManagement() {  
    printf("\n--- User Management ---\n");  
    printf("Managing users...\n");  
}
```