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

// Define structures for Policyholder, Policy, and Claim
typedef struct {
    int id;
    char name[50];
    char address[100];
    char phone[15];
} Policyholder;

typedef struct {
    int id;
    int policyholder_id;
    char coverage[100];
    float premium;
} Policy;

typedef struct {
    int id;
    int policy_id;
    char description[100];
    char status[20];
} Claim;

// Global arrays to store policyholders, policies, and claims
```

```
Policyholder policyholders[100];
Policy policies[100];
Claim claims[100];

int policyholder_count = 0;
int policy_count = 0;
int claim_count = 0;

// Function declarations
void addPolicyholder();
void viewPolicyholder();
void editPolicyholder();
void deletePolicyholder();
void addPolicy();
void viewPolicy();
void editPolicy();
void deletePolicy();
void addClaim();
void viewClaim();
void updateClaim();
void deleteClaim();
void menu();

int main() {
    menu();
    return 0;
}
```

```
void menu() {
    int choice;
    while(1) {
        printf("\nHealth Insurance Management System\n");
        printf("1. Add Policyholder\n");
        printf("2. View Policyholder\n");
        printf("3. Edit Policyholder\n");
        printf("4. Delete Policyholder\n");
        printf("5. Add Policy\n");
        printf("6. View Policy\n");
        printf("7. Edit Policy\n");
        printf("8. Delete Policy\n");
        printf("9. Add Claim\n");
        printf("10. View Claim\n");
        printf("11. Update Claim\n");
        printf("12. Delete Claim\n");
        printf("13. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        switch(choice) {
            case 1: addPolicyholder(); break;
            case 2: viewPolicyholder(); break;
            case 3: editPolicyholder(); break;
            case 4: deletePolicyholder(); break;
            case 5: addPolicy(); break;
            case 6: viewPolicy(); break;
            case 7: editPolicy(); break;
            case 8: deletePolicy(); break;
            case 9: addClaim(); break;
            case 10: viewClaim(); break;
            case 11: updateClaim(); break;
            case 12: deleteClaim(); break;
            case 13: exit(0);
            default: printf("Invalid choice!\n");
        }
    }
}

void addPolicyholder() {
    Policyholder p;
    p.id = policyholder_count + 1;
    printf("Enter name: ");
    scanf("%s", p.name);
    printf("Enter address: ");
    scanf("%s", p.address);
    printf("Enter phone: ");
    scanf("%s", p.phone);
    policyholders[policyholder_count++] = p;
    printf("Policyholder added successfully!\n");
}

void viewPolicyholder() {
    int id, i;

    printf("Enter Policyholder ID: ");

    scanf("%d", &id);

    for(i = 0; i < policyholder_count; i++) {

        if(policyholders[i].id == id) {

            printf("ID: %d\n", policyholders[i].id);

            printf("Name: %s\n", policyholders[i].name);

            printf("Address: %s\n", policyholders[i].address);

            printf("Phone: %s\n", policyholders[i].phone);

            return;

        }

    }

    printf("Policyholder not found!\n");

}


void editPolicyholder() {

    int id, i;

    printf("Enter Policyholder ID: ");

    scanf("%d", &id);

    for(i = 0; i < policyholder_count; i++) {

        if(policyholders[i].id == id) {

            printf("Enter new name: ");

            scanf("%s", policyholders[i].name);

            printf("Enter new address: ");

            scanf("%s", policyholders[i].address);

            printf("Enter new phone: ");

            scanf("%s", policyholders[i].phone);
            printf("Policyholder updated successfully!\n");
            return;
        }
    }
    printf("Policyholder not found!\n");
}

void deletePolicyholder() {
    int id, i, j;
    printf("Enter Policyholder ID: ");
    scanf("%d", &id);
    for(i = 0; i < policyholder_count; i++) {
        if(policyholders[i].id == id) {
            for(j = i; j < policyholder_count - 1; j++) {
                policyholders[j] = policyholders[j+1];
            }
            policyholder_count--;
            printf("Policyholder deleted successfully!\n");
            return;
        }
    }
    printf("Policyholder not found!\n");
}

void addPolicy() {
    Policy p;
    p.id = policy_count + 1;
    printf("Enter Policyholder ID: ");
    scanf("%d", &p.policyholder_id);
    printf("Enter coverage: ");
    scanf("%s", p.coverage);
    printf("Enter premium: ");
    scanf("%f", &p.premium);
    policies[policy_count++] = p;
    printf("Policy added successfully!\n");
}

void viewPolicy() {
    int id, i;
    printf("Enter Policy ID: ");
    scanf("%d", &id);
    for(i = 0; i < policy_count; i++) {
        if(policies[i].id == id) {
            printf("ID: %d\n", policies[i].id);
            printf("Policyholder ID: %d\n", policies[i].policyholder_id);
            printf("Coverage: %s\n", policies[i].coverage);
            printf("Premium: %.2f\n", policies[i].premium);
            return;
        }
    }
    printf("Policy not found!\n");
}

void editPolicy() {
    int id, i;

    printf("Enter Policy ID: ");

    scanf("%d", &id);

    for(i = 0; i < policy_count; i++) {

        if(policies[i].id == id) {

            printf("Enter new coverage: ");

            scanf("%s", policies[i].coverage);

            printf("Enter new premium: ");

            scanf("%f", &policies[i].premium);

            printf("Policy updated successfully!\n");

            return;

        }

    }

    printf("Policy not found!\n");

}


void deletePolicy() {

    int id, i, j;

    printf("Enter Policy ID: ");

    scanf("%d", &id);

    for(i = 0; i < policy_count; i++) {

        if(policies[i].id == id) {

            for(j = i; j < policy_count - 1; j++) {

                policies[j] = policies[j+1];

            }

            policy_count--;

            printf("Policy deleted successfully!\n");
            return;
        }
    }
    printf("Policy not found!\n");
}

void addClaim() {
    Claim c;
    c.id = claim_count + 1;
    printf("Enter Policy ID: ");
    scanf("%d", &c.policy_id);
    printf("Enter description: ");
    scanf("%s", c.description);
    printf("Enter status: ");
    scanf("%s", c.status);
    claims[claim_count++] = c;
    printf("Claim added successfully!\n");
}

void viewClaim() {
    int id, i;
    printf("Enter Claim ID: ");
    scanf("%d", &id);
    for(i = 0; i < claim_count; i++) {
        if(claims[i].id == id) {
            printf("ID: %d\n", claims[i].id);
            printf("Policy ID: %d\n", claims[i].policy_id);
            printf("Description: %s\n", claims[i].description);
            printf("Status: %s\n", claims[i].status);
            return;
        }
    }
    printf("Claim not found!\n");
}

void updateClaim() {
    int id, i;
    printf("Enter Claim ID: ");
    scanf("%d", &id);
    for(i = 0; i < claim_count; i++) {
        if(claims[i].id == id) {
            printf("Enter new status: ");
            scanf("%s", claims[i].status);
            printf("Claim updated successfully!\n");
            return;
        }
    }
    printf("Claim not found!\n");
}

void deleteClaim() {
    int id, i, j;
    printf("Enter Claim ID: ");
    scanf("%d", &id);
    for(i = 0; i < claim_count; i++) {

        if(claims[i].id == id) {

            for(j = i; j < claim_count - 1; j++) {

                claims[j] = claims[j+1];

            }

            claim_count--;

            printf("Claim deleted successfully!\n");

            return;

        }

    }

}
```

OUTPUT

```
Health Insurance Management System

1. Add Policyholder

2. View Policyholder

3. Edit Policyholder

4. Delete Policyholder

5. Add Policy

6. View Policy

7. Edit Policy

8. Delete Policy

9. Add Claim

10. View Claim

11. Update Claim
```

```
12. Delete Claim
13. Exit
Enter your choice: 1
Enter name: Ameya
Enter address: panmanayil
Enter phone: 9878675610
Policyholder added successfully!

Health Insurance Management System
1. Add Policyholder
2. View Policyholder
3. Edit Policyholder
4. Delete Policyholder
5. Add Policy
6. View Policy
7. Edit Policy
8. Delete Policy
9. Add Claim
10. View Claim
11. Update Claim
12. Delete Claim
13. Exit
Enter your choice: 2
Enter Policyholder ID: 1
ID: 1
Name: Ameya
Address: panmanayil
```

```
Phone: 9878675610


Health Insurance Management System
1. Add Policyholder
2. View Policyholder
3. Edit Policyholder
4. Delete Policyholder
5. Add Policy
6. View Policy
7. Edit Policy
8. Delete Policy
9. Add Claim
10. View Claim
11. Update Claim
12. Delete Claim
13. Exit
Enter your choice: 10
Enter Claim ID: 1
Claim not found!


Health Insurance Management System
1. Add Policyholder
2. View Policyholder
3. Edit Policyholder
4. Delete Policyholder
5. Add Policy
6. View Policy
```

```
7. Edit Policy
8. Delete Policy
9. Add Claim
10. View Claim
11. Update Claim
12. Delete Claim
13. Exit
Enter your choice: 9
Enter Policy ID: 1
Enter description: 10000
Enter status: added
Claim added successfully!

Health Insurance Management System
1. Add Policyholder
2. View Policyholder
3. Edit Policyholder
4. Delete Policyholder
5. Add Policy
6. View Policy
7. Edit Policy
8. Delete Policy
9. Add Claim
10. View Claim
11. Update Claim
12. Delete Claim
13. Exit
```

```
Enter your choice: 10
Enter Claim ID: 1
ID: 1
Policy ID: 1
Description: 10000
Status: added

Health Insurance Management System
1. Add Policyholder
2. View Policyholder
3. Edit Policyholder
4. Delete Policyholder
5. Add Policy
6. View Policy
7. Edit Policy
8. Delete Policy
9. Add Claim
10. View Claim
11. Update Claim
12. Delete Claim
13. Exit
Enter your choice:5
Enter Policyholder ID: 1
Enter coverage: 10000
Enter premium: 20000
Policy added successfully!
Health Insurance Management System
```

```
1. Add Policyholder
2. View Policyholder
3. Edit Policyholder
4. Delete Policyholder
5. Add Policy
6. View Policy
7. Edit Policy
8. Delete Policy
9. Add Claim
10. View Claim
11. Update Claim
12. Delete Claim
13. Exit
Enter your choice: 1
Enter name: Ameya
Enter address: panmanayil
Enter phone: 1234567890
Policyholder added successfully!

Health Insurance Management System
1. Add Policyholder
2. View Policyholder
3. Edit Policyholder
4. Delete Policyholder
5. Add Policy
6. View Policy
7. Edit Policy
```

```
8. Delete Policy
9. Add Claim
10. View Claim
11. Update Claim
12. Delete Claim
13. Exit
Enter your choice: 6
Enter Policy ID: 1
ID: 1
Policyholder ID: 1
Coverage: 10000
Premium: 20000.00
```