WEEK - 1

```c
#include <stdio.h>
#include <stdlib.h>
#define SIZE 12
int hash[SIZE] = {0};
void insertLinear(int data) {
    int index = data % SIZE;
    int i = 0;
    while (hash[(index + i) % SIZE] != 0) {
        i++;
    }
    hash[(index + i) % SIZE] = data;
}
int searchLinear(int data){
    int index = data % SIZE;
    int i = 0;
    while (hash[(index + i) % SIZE] != data) {
        if (hash[(index + i) % SIZE] == 0) {
            return -1;
        }
        i++;
    }
    return (index + i) % SIZE;
}
void insertQuadratic(int data){
    int index = data % SIZE;
    int i = 0,j=0;
    while (hash[(index + i * i) % SIZE] != 0) {
        i++;
    }
    hash[(index + i * i) % SIZE] = data;
}
int searchQuadratic(int data){
    int index = data % SIZE;
    int i = 0;
    while (hash[(index + i * i) % SIZE] != data){
        if (hash[(index + i * i) % SIZE] == 0){
            return -1;
        }
        i++;
```

```
    }
    return (index + i * i) % SIZE;
}
void display() {
    printf("Hash Table:\n");
    for (int i = 0; i < SIZE; i++){
        printf("%d\n ", hash[i]);
    }
    printf("\n");
}
int main(){
    int choice, data, index;
    printf("1.Insert using Linear Probing\n");
    printf("2.Search using Linear Probing\n");
    printf("3.Insert using Quadratic Probing\n");
    printf("4.Search using Quadratic Probing\n");
    printf("5.Display Hash Table\n");
    printf("6.Exit\n");
    while (1){
        printf("Enter your choice: ");
        scanf("%d", &choice);
        switch (choice) {
            case 1:
                printf("Enter the data to insert: ");
                scanf("%d", &data);
                insertLinear(data);
                break;
            case 2:
                printf("Enter the data to search: ");
                scanf("%d", &data);
                index = searchLinear(data);
                if (index == -1) {
                    printf("Data not found\n");
                } else{
                    printf("Data found at index %d\n", index);
                }
                break;
            case 3:
                printf("Enter the data to insert: ");
                scanf("%d", &data);
```

```
                insertQuadratic(data);
                break;
            case 4:
                printf("Enter the data to search: ");
                scanf("%d", &data);
                index = searchQuadratic(data);
                if (index == -1) {
                    printf("Data not found\n");
                } else {
                    printf("Data found at index %d\n", index);
                }
                break;
            case 5:
                display();
                break;
            case 6:
                exit(0);
            default:
                printf("Invalid choice\n");
        }
    }
    return 0;
}
```

Output:

```
Enter your choice: 1
Enter the data to insert: 12
Enter your choice: 1
Enter the data to insert: 4
Enter your choice: 1
Enter the data to insert: 5
Enter your choice: 1
Enter the data to insert: 7
Enter your choice: 1
Enter the data to insert: 16
Enter your choice: 1
Enter the data to insert: 15
Enter your choice: 5
Hash Table:
12
 0
 0
 15
 4
 5
 16
 7
 0
 0
 0
 0
```

```
1.Insert using Linear Probing
2.Search using Linear Probing
3.Insert using Quadratic Probing
4.Search using Quadratic Probing
5.Display Hash Table
6.Exit
Enter your choice: 3
Enter the data to insert: 10
Enter your choice: 3
Enter the data to insert: 4
Enter your choice: 3
Enter the data to insert: 12
Enter your choice: 3
Enter the data to insert: 16
Enter your choice: 3
Enter the data to insert: 5
Enter your choice: 3
Enter the data to insert: 15
Enter your choice: 5
Hash Table:
12
 0
 0
 15
 4
 16
 5
 0
 0
 0
 10
 0
```