```
//Created By Ritwik Chandra Pandey
//On 4th Nov
//Collision Resolution Techniques: Double Hashing

#include<stdio.h>
#include<conio.h>
#define SIZE 13
#define PRIME 7


int HashTable[SIZE];
int hash(int x) {
	return x % SIZE;
}
int hash2(int x) {
	return PRIME - x % PRIME;
}
void insert(int x) {
	int i,index, start;
	i=1;
	index = hash(x);
	start = index;
	while(HashTable[index]!=-1){
		if(HashTable[index]!=-1){
			break;
		}
		if(HashTable[index]==x){
			printf("%d already exists in the hash table.",x);
			return;
		}
		index = (start+i*hash2(x))%SIZE;
		i++;
		if(index!=start){
			printf("Hash table is full. So cannot insert the element.\n");
			return;

		}
	}
	HashTable[index]=x;
	printf("Successfully inserted.\n");
```

```
}
void delete(int x) {
	int i,index,start;
	i=1;
	index= hash(x);
	start = index;
	while(HashTable[index]!=x){
		if(HashTable[index]==x){
			break;
		}
		index = (start+ i*hash2(x))%SIZE;
		i++;
		if(index==start){
			printf("Element not found. So cannot delete the element.\n");
			return;
		}
	}
	HashTable[index]=-1;
	printf("Successfully deleted.\n");
}
void search(int x) {
	int i,index,start;
	i=1;
	index= hash(x);
	start = index;
	while(HashTable[index]!=x){
		if(HashTable[index]==x){
			break;
		}
		index = (start+i*hash2(x))%SIZE;
		i++;
		if(index==start){
			printf("Element not found.\n");
			return;
		}
	}
	printf("Element found.\n");
}
void print() {
	int i;
```

```
        for(i=0;i< SIZE;i++){
                if(HashTable[i]!=-1)
                  printf("[%d]=>%d ",i,HashTable[i]);
        }
        printf("\n");
}


int main() {
        int x, op, i=0;
        for(i=0;i<SIZE;i++)
                HashTable[i]=-1;
        while(1) {
                printf("1.Insert 2.Delete 3.Search 4.Print 5.Exit\n");
                printf("Enter your option : ");
                scanf("%d", &op);
                switch(op) {
                        case 1:printf("Enter an element to be inserted : ");
                                        scanf("%d", &x);
                                        insert(x);
                                        break;
                        case 2:
                                        printf("Enter an element to be deleted : ");
                                        scanf("%d", &x);
                                        delete(x);
                                        break;
                        case 3:
                                        printf("Enter an element to be searched : ");
                                        scanf("%d", &x);
                                        search(x);
                                        break;
                        case 4:
                                        print();
                                        break;
                        case 5:exit(0);
                }
        }
}
```