WAP for Hashing with Linear Probing

```c
#include <stdio.h>
#define SIZE 10

int hashFunction (int key)
{
    return key % SIZE;
}

void insertValue (int hashTable[], int key)
{
    int i=0;
    int hkey = hashFunction (key);
    int index;
    do {
        index = (hkey + i) % SIZE;
        if (hashTable [index] == -1) {
            hashTable [index] = key;
            printf ("inserted key %d at index %d \n", key, index);
            return;
        }
        i++;
    } while (i < SIZE);
    printf ("unable to insert key %d hash table is full \n", key);
}

int searchValue (int hashTable[], int key)
{
    int i=0;
    int hkey = hashFunction (key);
    int index;
    do {
        index = (hkey + i) % SIZE;
        if (hashTable [index]
        {
```

```c
        printf("key %d found at index %d\n", key, index);
        return index;
    }
    i++;
} while (i < SIZE);
printf("key %d not found in hash table\n", key);
return -1;
}


void main()
{
    int hashTable[SIZE];
    for (int i=0; i<SIZE; i++)
    {
        hashTable[i] = -1;
    }
    insertValue(hashTable, 1234);
    insertValue(hashTable, 5678);
    insertValue(hashTable, 9876);
    searchValue(hashTable, 5678);
    searchValue(hashTable, 1111);
    return 0;
}
```

S.T