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

int keys[20], hashIndex, numKeys, tableSize, *hashTable, elementCount = 0;
int i;


void creatHashTable() {
    hashTable = (int*)malloc(tableSize * sizeof(int));
    if (hashTable == NULL) {
        printf("MEMORY UNAVAILABLE:\n");
    } else {
        for (i = 0; i < tableSize; i++) {
            hashTable[i] = -1;
        }
    }
}


void insertIntoHashTable(int key) {
    hashIndex = key % tableSize;

    while (hashTable[hashIndex] != -1) {
        hashIndex = (hashIndex + 1) % tableSize;
    }
    hashTable[hashIndex] = key;
    elementCount++;
}


void displayHashTable() {
    int i;
    if (elementCount == 0) {
        printf("HASH TABLE EMPTY\n");
    }
    for (i = 0; i < tableSize; i++) {
        printf("T[%d]--> %d\n",i,hashTable[i]);
    }
}


int main (){
    int i;
    printf("Enter the number of keys: ");
    scanf("%d", &numKeys);
    printf("Enter the hash table size: ");
    scanf("%d", &tableSize);
    printf("Enter the keys: \n");
    for(i = 0; i < numKeys; i++) {
        scanf("%d", &keys[i]);
    }

    creatHashTable();

    printf("Inserting keys into the hash table:\n");
    for(i = 0; i < numKeys; i++) {
        if (elementCount == tableSize) {
            printf("HASH TABLE FULL\n");
            printf("Can't insert the key %d\n", keys[i]);
            break;
        }
        insertIntoHashTable(keys[i]);
    }

    displayHashTable();

    return 0;
}
```