#include <stdio.h>

#define SIZE 10

int hashTable[SIZE];

void initHashTable() {

for (int i = 0; i < SIZE; i++)

hashTable[i] = -1;

}

int hashFunction(int key) {

return key % SIZE;

}

void insert(int key) {

int index = hashFunction(key);

int i = 0;

while (hashTable[(index + i) % SIZE] != -1 && i < SIZE) {

i++;

}

if (i < SIZE) {

hashTable[(index + i) % SIZE] = key;

printf("Inserted %d at index %d\n", key, (index + i) % SIZE);

} else {

printf("Hash table is full. Cannot insert %d\n", key);

}

}

void display() {

printf("\nHash Table:\n");

for (int i = 0; i < SIZE; i++) {

if (hashTable[i] != -1)

printf("Index %d: %d\n", i, hashTable[i]);

else

printf("Index %d: empty\n", i);

}

}

int main() {

int choice, key;

initHashTable();

while (1) {

printf("\n1. Insert\n2. Display\n3. Exit\n");

printf("Enter choice: ");

scanf("%d", &choice);

switch (choice) {

case 1:

printf("Enter key to insert: ");

scanf("%d", &key);

insert(key);

break;

case 2:

display();

break;

case 3:

return 0;

default:

printf("Invalid choice\n");

}

}

}

