/\*

Implement all the functions of a dictionary (ADT) using hashing (Without Replacement).

Data: Set of (key, value) pairs, Keys are mapped to values, Keys must be comparable, Keys

must be unique Standard Operations: Insert(key, value), Find(key), Delete(key)

\*/

#include<iostream>

#include<string.h>

using namespace std;

class HashFunction

{

typedef struct hash

{

int key;

int value;

}hash;

hash h[10];

public:

HashFunction();

void insert();

void display();

int find(int);

void Delete(int);

};

HashFunction::HashFunction()

{

int i;

for(i=0;i<10;i++)

{

h[i].key=-1;

h[i].value=-1;

}

}

void HashFunction::Delete(int k)

{

int index=find(k);

if(index==-1)

{

cout<<"\n\tKey Not Found";

}

else

{

h[index].key=-1;

h[index].value=-1;

cout<<"\n\tKey is Deleted";

}

}

int HashFunction::find(int k)

{

int i;

for(i=0;i<10;i++)

{

if(h[i].key==k)

{

cout<<"\n\t"<<h[i].key<<" is Found at "<<i<<" Location With Value "<<h[i].value;

return i;

}

}

if(i==10)

{

return -1;

}

}

void HashFunction::display()

{

int i;

cout<<"\n\t\tKey\tValue";

for(i=0;i<10;i++)

{

cout<<"\n\th["<<i<<"]\t"<<h[i].key<<"\t"<<h[i].value;

}

}

void HashFunction::insert()

{

char ans;

int k,v,hi,cnt=0,flag=0,i;

do

{

if(cnt>=10)

{

cout<<"\n\tHash Table is FULL";

break;

}

cout<<"\n\tEnter a Key: ";

cin>>k;

cout<<"\n\tEnter a Value: ";

cin>>v;

hi=k%10;// hash function

if(h[hi].key==-1)

{

h[hi].key=k;

h[hi].value=v;

}

else

{

for(i=hi+1;i<10;i++)

{

if(h[i].key==-1)

{

h[i].key=k;

h[i].value=v;

flag=1;

break;

}

}

for(i=0;i<hi && flag==0;i++)

{

if(h[i].key==-1)

{

h[i].key=k;

h[i].value=v;

break;

}

}

}

flag=0;

cnt++;

cout<<"\n\t..... Do You Want to Insert More Key: ";

cin>>ans;

}while(ans=='y'||ans=='Y');

}

main()

{

int ch,k,index;

char ans;

HashFunction obj;

do

{

cout<<"\n\t\*\*\*\*\* Dictionary (ADT) \*\*\*\*\*";

cout<<"\n\t1. Insert\n\t2. Display\n\t3. Find\n\t4. Delete\n\t5. Exit";

cout<<"\n\t..... Enter Your Choice: ";

cin>>ch;

switch(ch)

{

case 1: obj.insert();

break;

case 2: obj.display();

break;

case 3: cout<<"\n\tEnter a Key Which You Want to Search: ";

cin>>k;

index=obj.find(k);

if(index==-1)

{

cout<<"\n\tKey Not Found";

}

break;

case 4: cout<<"\n\tEnter a Key Which You Want to Delete: ";

cin>>k;

obj.Delete(k);

break;

case 5:

break;

}

cout<<"\n\t..... Do You Want to Continue in Main Menu: ";

cin>>ans;

}while(ans=='y'||ans=='Y');

}

/\*

~$ cd Desktop/

/Desktop$ g++ hashd.cpp

/Desktop$ ./a.out

\*\*\*\*\* Dictionary (ADT) \*\*\*\*\*

1. Insert

2. Display

3. Find

4. Delete

5. Exit

..... Enter Your Choice: 1

Enter a Key: 65

Enter a Value: 5

..... Do You Want to Insert More Key: y

Enter a Key: 77

Enter a Value: 7

..... Do You Want to Insert More Key: y

Enter a Key: 55

Enter a Value: 5

..... Do You Want to Insert More Key: y

Enter a Key: 35

Enter a Value: 5

..... Do You Want to Insert More Key: y

Enter a Key: 99

Enter a Value: 9

..... Do You Want to Insert More Key: y

Enter a Key: 67

Enter a Value: 7

..... Do You Want to Insert More Key: n

..... Do You Want to Continue in Main Menu: y

\*\*\*\*\* Dictionary (ADT) \*\*\*\*\*

1. Insert

2. Display

3. Find

4. Delete

5. Exit

..... Enter Your Choice: 2

Key Value

h[0] 67 7

h[1] -1 -1

h[2] -1 -1

h[3] -1 -1

h[4] -1 -1

h[5] 65 5

h[6] 55 5

h[7] 77 7

h[8] 35 5

h[9] 99 9

..... Do You Want to Continue in Main Menu: y

\*\*\*\*\* Dictionary (ADT) \*\*\*\*\*

1. Insert

2. Display

3. Find

4. Delete

5. Exit

..... Enter Your Choice: 3

Enter a Key Which You Want to Search: 35

35 is Found at 8 Location With Value 5

..... Do You Want to Continue in Main Menu: y

\*\*\*\*\* Dictionary (ADT) \*\*\*\*\*

1. Insert

2. Display

3. Find

4. Delete

5. Exit

..... Enter Your Choice: 4

Enter a Key Which You Want to Delete: 35

35 is Found at 8 Location With Value 5

Key is Deleted

..... Do You Want to Continue in Main Menu: y

\*\*\*\*\* Dictionary (ADT) \*\*\*\*\*

1. Insert

2. Display

3. Find

4. Delete

5. Exit

..... Enter Your Choice: 2

Key Value

h[0] 67 7

h[1] -1 -1

h[2] -1 -1

h[3] -1 -1

h[4] -1 -1

h[5] 65 5

h[6] 55 5

h[7] 77 7

h[8] -1 -1

h[9] 99 9

..... Do You Want to Continue in Main Menu:

\*/