/\*Implement all the functions of a dictionary (ADT) using hashing and handle collisions using chaining with / 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>

#include<stdlib.h>

using namespace std;

struct data

{ char name[30];

char name1[30];

};

class hash

{ int n,sum,x,c,i,j; char na[30],na1[30];

data d[10];

public:

hash()

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

{ strcpy(d[i].name,"\0"); }

}

void insert();

void search();

void delet();

void display();

};

void hash::insert()

{

cout<<"\n enter no. of words";

cin>>n;

for(j=0;j<n;j++)

{ cout<<"\n\n enter the word";

cin>>na;

cout<<"\n enter the meaning of that word";

cin>>na1;

sum=0;

for(i=0;i<strlen(na);i++)

{ sum=sum+na[i];

}

x=(sum/strlen(na))%10;

cout<<x;

c=x;

while(1)

{

if(!strcmp(d[x].name,"\0"))

{ strcpy(d[x].name,na);

strcpy(d[x].name1,na1);

break;

}

x=(x+1)%10;

if(c==x)

{ cout<<"\n hash table is full";

break;

}

}

}

}

void hash::search()

{ cout<<"\n enter the word whose meaning you want";

cin>>na;

sum=0;

for(i=0;i<strlen(na);i++)

{ sum=sum+(int)na[i];

}

x=(sum/strlen(na))%10;

c=x;

while(1)

{

if(!strcmp(d[x].name,na))

{ cout<<"\n MEANING-> "<<d[x].name<<"="<<d[x].name1;

break;

}

x=(x+1)%10;

if(c==x)

{ cout<<"\n word not found";

break;

}

}

}

void hash::delet()

{ cout<<"\n enter the word which is to be deleted";

cin>>na;

sum=0;

for(i=0;i<strlen(na);i++)

{ sum=sum+(int)na[i];

}

x=(sum/strlen(na))%10;

c=x;

while(1)

{

if(!strcmp(d[x].name,na))

{ cout<<"\n"<<d[x].name<<" word deleted";

strcpy(d[x].name,"\0"); strcpy(d[x].name1,"\0");

break;

}

x=(x+1)%10;

if(c==x)

{ cout<<"\n word not found";

break;

}

}

}

void hash::display()

{

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

cout<<endl<<d[i].name<<" "<<d[i].name1;

}

}

int main()

{

hash h; int n;

while(1)

{

cout<<"\n enter the choice";

cout<<"\n 1.insert word and its meaning";

cout<<"\n 2.find meaning";

cout<<"\n 3.delete the word";

cout<<"\n 4.exit";

cin>>n;

switch(n)

{

case 1: h.insert();

break;

case 2: h.search();

break;

case 3: h.delet();

break;

case 4: exit(0);

default: cout<<"\n unknown choice";

}

}

return 0;

}