**南京信息工程大学 实验（实习）报告**

实验（实习）名称 实验七 日期 2021/12/19 得分 指导教师 陈金辉

学院 计算机与软件 专业信息安全 年级 2020 班次 3 姓名 沈宇豪 学号 202083290386

1. 实验内容

基本要求

假设人名为中国人姓名的汉语拼音形式。待填入哈希表的人名共有30个，取平均查找长度的上限为2。哈希函数用除留余数法构造，用伪随机探测再散列法处理冲突。

2．实验步骤和实验结果

#include<stdio.h>

#include<time.h>

#include<stdlib.h>

#include<string.h>

#include<ctype.h>

#define hash\_len 50

#define P 47

#define name\_len 30

typedef struct //姓名表

{

char \*key;

int m;

}NAME;

typedef struct

{

char \*key;

int m; //关键字

int si; // 查找长度

}HASH;

NAME Name[hash\_len];//全局定义姓名表

HASH Hash[hash\_len]; //全局定义哈希表

int d[30],i,j;

void InitName()

{

Name[0].key="theshy";

Name[1].key="ning";

Name[2].key="rookie";

Name[3].key="jklove";

Name[4].key="baolan";

Name[5].key="tian";

Name[6].key="doinb";

Name[7].key="lwx";

Name[8].key="lucas";

Name[9].key="xun";

Name[10].key="zoom";

Name[11].key="Ale";

Name[12].key="jimgong";

Name[13].key="zoom";

Name[14].key="ming";

Name[15].key="jiejie";

Name[16].key="scout";

Name[17].key="bin";

Name[18].key="hangfeng";

Name[19].key="icon";

Name[20].key="xiaohu";

Name[21].key="gala";

Name[22].key="flandre";

Name[23].key="nuguri";

Name[24].key="wei";

Name[25].key="tarzan";

Name[26].key="viper";

Name[27].key="mekio";

Name[28].key="uzi";

Name[29].key="faker";

for(i=0;i<name\_len;i++)

{

int s=0;

char \*p=Name[i].key;

for(j=0;\*(p+j)!='\0';j++)

{

s+=toascii(\*(p+j));

Name[i].m=s;

}

}

}

void createhash()

{

for(i=0;i<hash\_len;i++)

{

Hash[i].key="\0";

Hash[i].m=0;

Hash[i].si=0;

}

for(i=0;i<name\_len;i++)

{

int sum=1,j=0,t;

int adr=(Name[i].m)%P;

if(Hash[adr].si==0)

{

Hash[adr].m=Name[i].m;

Hash[adr].key=Name[i].key;

Hash[adr].si=1;

}

else

{

t=adr;

for(;Hash[adr].si!=0&&adr<hash\_len;adr++)

{

sum=sum+1;

if(adr==hash\_len-1)

{

for(;Hash[adr].si!=0&&adr<t;adr++)

sum=sum+1;

if(adr==t)

printf("哈希表已满\n");

}

}

Hash[adr].m=Name[i].m;

Hash[adr].key=Name[i].key;

Hash[adr].si=sum;

}

}

}

void displayname()

{

printf("\n地址 \t\t 姓名 \t\t 关键字\n");

for(i=0;i<name\_len;i++)

printf("%2d %18s \t\t %d \n",i,Name[i].key,Name[i].m);

}

void displayhash()

{

float asl=0.0;

printf("\n\n 地址 \t\t 姓名 \t\t 关键字 \t 搜索长度\n") ;

for(i=0;i<hash\_len;i++)

{

printf("%2d %18s \t\t %d \t\t %d\n",i,Hash[i].key,Hash[i].m,Hash[i].si);

asl+=Hash[i].si;

}

asl/=name\_len;

printf("\n\n平均查找长度：ASL(%d)=%f \n",name\_len,asl);

}

void findname()

{

char name[20]={0};

int s=0,sum=1,adr;

printf("\n请输入想要查找的姓名的拼音:");

scanf("%s",name);

getchar();

for (j=0;j<20;j++)//求出姓名的拼音所对应的ASCII作为关键字

s+=toascii(name[j]);

adr=s%P; //除留余数法

j=0;

if(Hash[adr].m==s&&!strcmp(Hash[adr].key,name)) //分3种情况进行判断，并输出查找结果

printf("\n姓名:%s 关键字:%d 地址:%d 查找长度为: 1\n",Hash[adr].key,s,adr);

else if (Hash[adr].m==0)

printf("\n没有想要查找的人!\n");

else

{

while(1)

{

adr=(adr+d[j++])%hash\_len;//伪随机探测再散列法处理冲突

sum=sum+1; //查找次数加1

if(Hash[adr].m==0)

{

printf("\n没有想要查找的人!\n");

break;

}

if(Hash[adr].m==s&&!strcmp(Hash[adr].key,name))

{

printf("\n姓名:%s 关键字:%d 地址:%d 查找长度为:%d\n",Hash[adr].key,adr,s,sum);

break;

}

}

}

}

void view()

{

printf("=======================================================\n");

printf("= =\n");

printf("= 哈希表 =\n");

printf("= =\n");

printf("= A: 打印姓名表 B: 打印哈希表 =\n");

printf("= =\n");

printf("= C: 查找 D: 退出程序 =\n");

printf("= =\n");

printf("=======================================================\n");

}

int main()

{

char c;

int a=1;

srand((int)time(0));//以当前时间对应的int值为随机序列起点，每次运行程序由于起点不同可以得到不同的随机数序列

for(i=0;i<30;i++)//用随机函数求得伪随机数列d[i]（在1到50之间）

d[i]=1+(int)(hash\_len\*rand()/(RAND\_MAX+1.0));

InitName();//调用初始化姓名表函数

createhash();//调用创建哈希表函数

view();

while(a)

{

printf("\n输入选项:");

scanf("%c",&c);

getchar();

switch(c)

{

case 'A':

case 'a': displayname(); break;//打印姓名表

case 'B':

case 'b': displayhash(); break;//打印哈希表

case 'C':

case 'c': findname(); break;//调用查找函数

case 'D':

case 'd': a=0; break;//退出循环，终止程序

default: printf("\n请输入正确的选择!\n"); break;

}

}

return 0;

}

3．分析与讨论



