1、/\*编程程序：完成sort函数，实现10个任意整数数据的升序输出。

输入数据：10 9 8 7 6 5 4 3 2 1

输出数据：the result data is :1 2 3 4 5 6 7 8 9 10

\*/

#include<stdio.h>

int main()

{

int a[10],i;

void sort(int \*,int);

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

scanf("%d",&a[i]);

sort(a,10);

printf("\nthe result data is :");

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

printf("%d ",a[i]);

return 0;

}

void sort(int \*p,int n)

{

int i,j,tmp;

for(i=0;i<n-1;i++)

for(j=i+1;j<n;j++)

if (p[i]>p[j])

{tmp=p[i];p[i]=p[j];p[j]=tmp;}

}

2、/\*编写程序：完成connect\_string函数，实现字符串的连接功能。

输入数据：

I am a Chinese,

输出数据：

I am a Chinese,I love my motherland!\*/

#include<stdio.h>

int main()

{

char a[100],b[100]="I love my motherland!";

void connect\_string(char \*,char \*);

gets(a);

connect\_string(a,b);

puts(a);

return 0;

}

void connect\_string(char \*p,char \*q)

{

char \*t;

t=p;

while (\*t!='\0') t++;

for(;\*q!='\0';q++)

\*t++=\*q;

\*t='\0';

}

3、/\*编写函数，统计任意一个字符串中哪个数字字符出现频率最高。

输入数据：abc12345689def9876ghi981jk8

输出数据：出现频率最高的数字字符是:8

\*/

#include <stdio.h>

#include <string.h>

char count(char \*st)

{

int j=0,max,k;

int tt[100]={0};

for(;\*st!='\0';st++)

if ((\*st>='0')&&(\*st<='9'))

tt[\*st-48]++;

max=tt[0];k=0;

for(j=1;j<10;j++)

if (max<tt[j]) {max=tt[j];k=j;}

return k+48;

}

int main()

{

char ss[80],ch;

gets(ss);

ch=count(ss);

printf("出现频率最高的数字字符是:");

printf("%c\n",ch);

return 0;

}

4、/\*编写函数，统计任意一个密码电文中英文字母出现了多少次。

输入数据：abc12345689def9876ghi981jk8

输出数据：英文字母出现的次数为:11

\*/

#include <stdio.h>

#include <string.h>

int count(char \*s)

{

int cnt=0;

for(;\*s!='\0';s++)

if ((\*s>='a')&&(\*s<='z')||(\*s>='A')&&(\*s<='Z'))

cnt++;

return cnt;

}

int main()

{

char ss[80];

gets(ss);

printf("英文字母出现的次数为:");

printf("%d",count(ss));

printf("\n");

return 0;

}

5、/\*完成crypt函数，实现加密并输出功能。

加密规则：字符串中所有小写英文字母循环加密。如a到b，b到c，…,z到a。

输入数据：zy have a little apple!

输出数据：the resulted data is :az ibwf b mjuumf bqqmf!

\*/

#include <stdio.h>

#include <string.h>

void crypt(char \*s)

{

while (\*s)

{

if (\*s>='a'&& \*s<='z')

if (\*s=='z') \*s='a';

else

\*s=\*s+1;

s++;

}

}

int main()

{

char t1[80];

gets(t1);

crypt(t1);

printf("\nthe resulted data is :%s",t1);

printf("\n");

return 0;

}

6、/\*编写程序：完成fun函数，实现将任意一个字符串中的原音字母抽取出来组成一个新的字符串并输出。

输入数据：I am a teacher,I am 39 years old.

输出数据：IaaeaeIaeao

\*/

#include<stdio.h>

void fun(char a[],char b[])

{

int i,j;

for(i=0,j=0;a[i]!='\0';i++)

if (a[i]=='a'||a[i]=='A'||a[i]=='e'||a[i]=='E'||a[i]=='i'||a[i]=='I'||a[i]=='o'||a[i]=='O'||a[i]=='u'||a[i]=='U')

b[j++]=a[i];

}

int main()

{ char a[80],b[80]={0};

gets(a);

fun(a,b);

puts(b);

return 0;

}

7、/\*编写函数：移动字符串中的内容，移动规则：把1到第m个字符，平移到字符串的最后，把第m+1到最后的字符移到字符串的前部。

输入数据：4

输出数据：mingbusida

\*/

#include <stdio.h>

#include <string.h>

#define N 80

void fun1(char \*w)

{

int i;char t;

t=w[0];

for(i=0;i<(int)strlen(w)-1;i++)

w[i]=w[i+1];

w[strlen(w)-1]=t;

}

void fun(char \*w,int m)

{int i;

for(i=1;i<=m;i++)

fun1(w);

}

int main()

{

char a[N]="sidamingbu";

int m;

scanf("%d",&m);

fun(a,m);

puts(a);

printf("\n");

return 0;

}

8、/\*利用递归方法求10个任意整数实数的最大值。

输入数据：1 2 3 4 5 6 7 8 10 9

输出数据：10

\*/

#include<stdio.h>

void main()

{

int i,a[10];

int aver(int [],int);

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

scanf("%d",&a[i]);

printf("%d\n",aver(a,10));

}

int aver(int a[],int n)

{

int c;

if (n==1) c=a[0];

else

{

if (aver(a,n-1)>a[n-1])

c=aver(a,n-1);

else

c=a[n-1];

}

return c;

}

9、/\*编程：完成avgint函数，利用递归法求一批数据的平均值。

输入数据：1 2 3 15 5 20 7 8 45 10

输出数据：the result data is :11.60

\*/

#include <stdio.h>

float avgint(int a[],int n)

{

float tt;

if (n==1) tt=a[0];

else

{

float mm;

mm=avgint(a,n-1);

tt=(mm\*(n-1)+a[n-1])/n;

}

return tt;

}

int main()

{

int a[10],i;

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

scanf("%d",&a[i]);

printf("the result data is :");

printf("%.2f\n",avgint(a,10));

return 0;

}

10、/\*买卖提将养的一缸金鱼分n次出售系统上一次卖出全部的一半加二分之一条；

第二次卖出余下的三分之一加三分之一条；

第三次卖出余下的四分之一加四分之一条；第四次卖出余下的五分之一加五分之一条；

最后卖出余下的11条。问原来的鱼缸中共有几条金鱼？

完成递归函数count2，计算鱼缸中的鱼的条数。

输入数据：5

输出数据：There are 59 fishes at first.

\*/

#include<stdio.h>

int count2(int n,int m)

{

int c;

if (n==m) c=11;

else

c=(count2(n+1,m)\*(n+1)+1)/n;

return c;

}

int main()

{

int x;

int m;

scanf("%d",&m);

x=count2(1,m);

printf("There are %d fishes at first.\n",x);

return 0;

}

11、/\*编写函数，将字符串中下标为偶数的字符按ASCII码升序排列，并将排序后下标为偶数的字符在主函数中输出。

输入数据：baawrskjghzlicda

输出数据：the result data is :abdgikrz

\*/

#include <stdio.h>

void fun(char st[],char tt[])

{

int i,j=0,k;

char ch;

for(i=0;st[i]!='\0';i++) //把奇数位置上的所有字符取出来

if (i%2==0) tt[j++]=st[i];

tt[j]=0;

//升序排列

for(i=0;i<j-1;i++)

for(k=i+1;k<j;k++)

if (tt[i]>tt[k]) {ch=tt[i];tt[i]=tt[k];tt[k]=ch;}

}

int main()

{

char st[80],tt[80];

gets(st);

fun(st,tt);

printf("the result data is :");

puts(tt);

return 0;

}

12、/\*编程：完成fun函数，实现将s所指字符串中的所有数字字符移到所有非数字字符之后，并保持数字字符和非数字字符串原有的先后次序。

输入数据：def35adh3kjsdf7

输出数据：defadhkjsdf3537

\*/

#include <stdio.h>

void fun(char s[])

{

int i,j=0,k=0;

char t1[80],t2[80];

for(i=0;s[i]!='\0';i++)

if (s[i]>='0'&&s[i]<='9')

{ t2[j]=s[i];j++; }

else t1[k++]=s[i];

t2[j]=0;t1[k]=0;

for(i=0;i<k;i++) s[i]=t1[i];

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

s[k+i]=t2[i];

}

int main()

{

char s[80];

gets(s);

fun(s);

printf("%s\n",s);

return 0;

}

13、/\*

编程：完成fun函数，除了尾部的\*和前导\*外，将字符串中的其余\*号全部删除，形参p已经指向字符串中最后的一个字母，不得使用c语言提供的字符串函数。前导和尾部的\*号不删除.

输入数据：\*\*\*\*\*\*\*AS\*BHH\*\*G\*\*\*\*\*\*\*

输出数据：\*\*\*\*\*\*\*ASBHHG\*\*\*\*\*\*\*

\*/

#include <stdio.h>

void fun(char \*a,char \*h,char \*p)

{

a=h;

for(;h<p;h++)

if (\*h!='\*') \*(a++)=\*h;

for(;\*h!='\0';h++)

\*(a++)=\*h;

\*a='\0';

}

int main()

{

char s[81],\*t,\*f;

gets(s);

t=f=s;

while (\*t) t++;

t--;

while (\*t=='\*') t--;

while (\*f=='\*') f++;

fun(s,f,t);

puts(s);

printf("\n");

return 0;

}

14、/\*编写函数：将ss所指字符串中所有下标为偶数位置上的字母转换为大写（若该位置上不是字母，则不转换）。

输入数据：I love my motherland!

输出数据：I LoVe mY MoThErLaNd!

\*/

#include <stdio.h>

#include <string.h>

#define MAX 100

void fun(char \*ss)

{

char \*t;

t=ss;

while (\*t)

{if ((\*t>='a')&&(\*t<='z')) \*t=\*t-32;

t=t+2;}

}

int main()

{

char tt[51];

gets(tt);

fun(tt);

printf("%s\n",tt);

return 0;

}

15、/\*

编程：删除一个升序排列的二维数组中重复的数字，重复数字只保留一个。

输入数据：2 2 3 4 4 5 6 6 6 7 7 8

输出数据：2 3 4 5 6 7 8

\*/

#include <stdio.h>

void fun(int \*tt,int n,int \*k)

{

int b[12],j=0,i;

//依次取a数组中的值，判断和b数组尾部元素值是否相等，把不相等者添加到b数组中

b[0]=tt[0];

for(i=1;i<n;i++)

if (tt[i]!=b[j])

b[++j]=tt[i];

//把整理后的数据重新写回到原始数组中

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

tt[i]=b[i];

\*k=j;

}

int main()

{

int a[12],i,n;

for(i=0;i<12;i++)

scanf("%d",&a[i]);

fun(a,12,&n);

for(i=0;i<n;i++) printf("%d ",a[i]);

printf("%d",a[n]);

printf("\n");

return 0;

}

16、/\*买卖提将养的一缸金鱼分n次出售,第一次卖出全部的一半加二分之一条；

第二次卖出余下的三分之一加三分之一条；

第三次卖出余下的四分之一加四分之一条；第四次卖出余下的五分之一加五分之一条；

最后卖出余下的11条。问原来的鱼缸中共有几条金鱼？

输入数据：5

输出数据：59

\*/

#include<stdio.h>

int count(int n)

{

int i,t=11;

for(i=n-1;i>=1;i--)

t=((i+1)\*t+1)/i;

return t;

}

int main()

{

int n,x;

//输入出售次数

scanf("%d",&n);

x=count(n);

//输出最初鱼缸中的鱼的条数

printf("%d\n",x);

return 0;

}

17、/\*编程：完成set\_grade函数，实现确定10个学生的成绩等级和不及格人数。

等级：A ：85－100；B：70－84；C：60－69；D：0－59

输入数据：zzs 88 zy 67 sl 55 jm 90 ly 56 hj 75 lz 90 wf 69 mt 79 hj 100

输出数据： zzs A zy C sl D jm A ly D hj B lz A wf C mt B hj A 2

\*/

#include <stdio.h>

struct student{

char name[20];

int score;

char grade;

};

typedef struct student st;

void inputstudent(st \*stu,int n)

{

int i;

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

{

scanf("%s",stu[i].name);

scanf("%d",&stu[i].score);

}

}

int set\_grade(struct student \* p,int n)

{

int i, k = 0;

for(i = 0; i <n; i++, p++){

if(p->score >= 85)

p->grade = 'A';

else if(p->score >= 70)

p->grade = 'B';

else if(p->score >= 60)

p->grade = 'C';

else{

p->grade = 'D';

k++;

}

}

return k;

}

void outputstudent(st \*p,int n)

{

int i;

for(i=0;i<n;i++,p++)

{

printf(" %s %c",p->name,p->grade);

}

}

int main(void)

{

struct student stu[10], \*ptr;

int count;

ptr = stu;

inputstudent(ptr,10);

count = set\_grade(ptr,10);

outputstudent(ptr,10);

printf(" %d\n",count);

return 0;

}

18、/\*

编程：现有一个班10个学生（姓名，成绩），完成fun函数，求出平均分并输出。

输入数据：85 76 69 85 91 72 64 87 58 78

输出数据：ave=76.50

\*/

#include <stdio.h>

#include <stdlib.h>

#define N 10

struct person

{

double s;

struct person \*next;

};

typedef struct person student;

double fun(student \*h)

{

double sum=0;

student \*p;

p=h->next;

while (p!=NULL)

{

sum=sum+p->s;

p=p->next;

}

return sum/N;

}

//建立班级信息链表

student \*creat()

{student \*h,\*p,\*q;

int i=0;

h=p=(student \*)malloc(sizeof(student));

while (i<N)

{

q=(student\*)malloc(sizeof(student));

scanf("%lf",&q->s);

i++;

p->next=q;

p=q;

}

p->next=NULL;

return h;

}

int main()

{

double ave;

student \*h;

h=creat();

ave=fun(h);

printf("ave=%.2f\n",ave);

printf("\n");

return 0;

}

19、/\*

编程：现有一个班10个学生（姓名，成绩），完成fun函数，求出最高分并输出。

输入数据：85 76 69 85 91 72 64 87 58 78

输出数据：max=91.00

\*/

#include <stdio.h>

#include <stdlib.h>

#define N 10

struct person

{

double s;

struct person \*next;

};

typedef struct person student;

double fun(student \*h)

{

double max;

student \*p;

p=h->next;

max=p->s;

while (p!=NULL)

{

if (p->s>max) max=p->s;

p=p->next;

}

return max;

}

//建立班级信息链表

student \*creat()

{student \*h,\*p,\*q;

int i=0;

h=p=(student \*)malloc(sizeof(student));

while (i<N)

{

q=(student\*)malloc(sizeof(student));

scanf("%lf",&q->s);

i++;

p->next=q;

p=q;

}

p->next=NULL;

return h;

}

int main()

{

double max;

student \*h;

h=creat();

max=fun(h);

printf("max=%.2f\n",max);

printf("\n");

return 0;

}

20、/\*

编程：现有一个班10个学生（姓名，成绩），完成fun函数，求出不及格的人数并输出。

输入数据：85 76 69 55 91 72 64 87 58 78

输出数据：persons=2

\*/

#include <stdio.h>

#include <stdlib.h>

#define N 10

struct person

{

double s;

struct person \*next;

};

typedef struct person student;

int fun(student \*h)

{

int cnt=0;

student \*p;

p=h->next;

while (p!=NULL)

{

if (p->s<60) cnt++;

p=p->next;

}

return cnt;

}

//建立班级信息链表

student \*creat()

{student \*h,\*p,\*q;

int i=0;

h=p=(student \*)malloc(sizeof(student));

while (i<N)

{

q=(student\*)malloc(sizeof(student));

scanf("%lf",&q->s);

i++;

p->next=q;

p=q;

}

p->next=NULL;

return h;

}

int main()

{

int n;

student \*h;

h=creat();

n=fun(h);

printf("persons=%d\n",n);

printf("\n");

return 0;

}

21、/\*编程：完成subtract函数，实现对计算任意两个24小时制的时间差。

输入数据：11:12:23 10:10:18

输出数据：1:2:5

\*/

#include <stdio.h>

#include <string.h>

typedef struct

{

int second;

int minute;

int hour;

}Time;

void swap(Time \*time1,Time \*time2)

{

//确保第一个时间比第二个时间大

if((time2->hour>time1->hour)||((time2->hour==time1->hour)&&(time2->minute>time1->minute))||((time2->hour==time1->hour)&&(time2->minute==time1->minute)&&(time2->second>time1->second)))

{

Time tmp;

tmp=\*time1;

\*time1=\*time2;

\*time2=tmp;

}

}

Time subtract(Time time1,Time time2)

{

Time result;

if (time1.second>=time2.second)

{

result.second=time1.second-time2.second;

}

else

{

time1.minute=time1.minute-1;

time1.second=time1.second+60;

result.second=time1.second-time2.second;

}

if (time1.minute>=time2.minute)

{

result.minute=time1.minute-time2.minute;

}

else

{

time1.hour=time1.hour-1;

time1.minute=time1.minute+60;

result.minute=time1.minute-time2.minute;

}

result.hour=time1.hour-time2.hour;

return result;

}

void printtime(Time time)

{

printf("%d:%d:%d",time.hour,time.minute,time.second);

}

int main()

{

Time time1;

Time time2;

Time time3;

char ch;

printf("输入两个时间：");

scanf("%d:%d:%d %d:%d:%d",&time1.hour,&time1.minute,&time1.second,&time2.hour,&time2.minute,&time2.second);

swap(&time1,&time2);

time3=subtract(time1,time2);

printtime(time3);

printf("\n");

return 0;

}

22、/\*

一批20本图书（书名，作者，出版社，价格，出版日期）。

编程：完成count函数，实现求购买某个出版社的书的册数。

输入数据：高等教育出版社

输出数据：7

\*/

#include <stdio.h>

#include <string.h>

struct cbrqs

{

int year;

int month;

int day;

};

typedef struct books

{

char name[30];

char author[20];

char publish[40];

float price;

struct cbrqs cbrq;

}book;

//求册数

int count(book tt[],int n,char ts[])

{

int i,cnt=0;

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

if (strcmp(tt[i].publish,ts)==0) cnt++;

return cnt;

}

int main()

{ book ts[20]={"操作系统","汤子瀛","北京大学出版社",37,1997,3,15,"操作系统","张丹","清华大学出版社",35,1999,5,40,"高级语言程序设计","谭浩强","高等教育出版社",40,1999,4,20,"程序设计基础","张琳","清华大学出版社",33,1994,5,15,"计算机基础","王楠","高等教育出版社",34,1995,8,15,"数据结构","严蔚敏","清华大学出版社",38,1996,4,25,"网络原理","张浩","高等教育出版社",39.5,1997,5,15,"高级语言程序设计","叶坤","北京大学出版社",36,1997,5,15,"操作系统","汤和","吉林大学出版社",35.5,1997,6,15,"java程序设计","张瀛","高等教育出版社",37,1993,3,15,"windows程序设计","苏芒","高等教育出版社",36.5,2000,7,27,"信息管理","林星","北京大学出版社",38,1998,9,12,"c程序设计","汤子瀛","北京大学出版社",37,1997,3,15,".net程序设计","何莎","高等教育出版社",37,1997,9,15,"操作系统-2","李刚","北京大学出版社",43,1999,6,11,"物联网基础","凌云","高等教育出版社",37,2017,5,29,"大数据处理","谭峰","北京大学出版社",37,2000,7,15,"操作系统","汤子","北京大学出版社",40.5,2004,3,15,"高级语言程序设计","张岩","吉林大学出版社",35,2010,12,18,"python程序设计","郑楚","北京大学出版社",35.5,2008,11,26};

int cs;

char cbs[80];

gets(cbs);

cs=count(ts,20,cbs);

printf("%d\n",cs);

return 0;

}

23、/\*设有三个候选人（库里，哈登，阿德托昆博），总共有50张选票，完成count函数，统计指定候选人获得的名次。

输入数据：库里

输出数据：第1名

\*/

#include <stdio.h>

#include <string.h>

struct person

{ char name[20];

int cnt;

};

int count(struct person st[],char \*tt[],char hxr[])

{

int i,j;

struct person tmp;

for(i=0;i<50;i++)

for(j=0;j<3;j++)

if (strcmp(st[j].name,tt[i])==0) st[j].cnt++;

for(i=0;i<2;i++)

for(j=i+1;j<3;j++)

if (st[i].cnt<st[j].cnt) {tmp=st[i];st[i]=st[j];st[j]=tmp;}

for(i=0;i<3;i++)

if (strcmp(st[i].name,hxr)==0)

break;

return i+1; }

int main()

{

char \*xp[50]={"库里","哈登","阿德托昆博","阿德托昆博","库里","库里","阿德托昆博","库里","阿德托昆博","库里","库里","库里","库里","库里","库里","库里","库里","库里","库里","库里","库里","库里","阿德托昆博","阿德托昆博","哈登","阿德托昆博","阿德托昆博","库里","库里","哈登","库里","哈登","库里","库里","库里","库里","库里","哈登","哈登","哈登","哈登","库里","库里","哈登","哈登","哈登","库里","库里","库里","库里"};

struct person leader[3]={"库里",0,"哈登",0,"阿德托昆博",0};

int n;

char hxr[20];

gets(hxr);

n=count(leader,xp,hxr);

printf("第%d名\n",n);

return 0;

}

24、/\*

10家公司（AA,BB,CC,DD,EE,FF,GG,HH,II,JJ,KK）投标,每个公司报一次价格，专家经过评标，选出中标公司。

评标规则：最接近平均报价的公司中标。

编程：实现自动评标。

输入数据：76 80 70 83 85 89 88 100 93 99

输出数据：中标的公司为：EE

\*/

#include <stdio.h>

#include <math.h>

struct com

{

char company[20];

int price;

};

typedef struct com com;

com fun(com \*t1,int n)

{

int i;

com tmp;

float sum=0,cj;

//计算平均报价

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

sum=sum+t1[i].price;

sum=sum/n;

tmp=t1[0];

cj=fabs(t1[0].price-sum);

for(i=1;i<n;i++)

if (fabs(t1[i].price-sum)<cj) {cj=fabs(t1[i].price-sum);tmp=t1[i];}

//返回中标公司编号

return tmp;

}

int main()

{

com gs[10]={"AA",0,"BB",0,"CC",0,"DD",0,"EE",0,"FF",0,"GG",0,"HH",0,"II",0,"JJ",0};

int i;

com zb;

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

scanf("%d",&gs[i].price);

zb=fun(gs,10);

printf("中标的公司为：%s\n",zb.company);

return 0;

}

25、/\*

一批20本图书（书名，作者，出版社，价格，出版日期）。

编程：完成count函数，实现求在某年(包括该年)以后购买书的费用。

输入数据：2000

输出数据：2000以后出版的书的成本为：221.50元

\*/

#include <stdio.h>

#include <string.h>

struct cbrqs

{

int year;

int month;

int day;

};

typedef struct books

{

char name[30];

char author[20];

char publish[40];

float price;

struct cbrqs cbrq;

}book;

//求费用

float count(book tt[],int n,int x)

{

int i;

float sum=0;

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

if (tt[i].cbrq.year>=x)

sum=sum+tt[i].price;

return sum;

}

int main()

{ book ts[20]={"操作系统","汤子瀛","北京大学出版社",37,1997,3,15,"操作系统","张丹","清华大学出版社",35,1999,5,40,"高级语言程序设计","谭浩强","高等教育出版社",40,1999,4,20,"程序设计基础","张琳","清华大学出版社",33,1994,5,15,"计算机基础","王楠","高等教育出版社",34,1995,8,15,"数据结构","严蔚敏","清华大学出版社",38,1996,4,25,"网络原理","张浩","高等教育出版社",39.5,1997,5,15,"高级语言程序设计","叶坤","北京大学出版社",36,1997,5,15,"操作系统","汤和","吉林大学出版社",35.5,1997,6,15,"java程序设计","张瀛","高等教育出版社",37,1993,3,15,"windows程序设计","苏芒","高等教育出版社",36.5,2000,7,27,"信息管理","林星","北京大学出版社",38,1998,9,12,"c程序设计","汤子瀛","北京大学出版社",37,1997,3,15,".net程序设计","何莎","高等教育出版社",37,1997,9,15,"操作系统-2","李刚","北京大学出版社",43,1999,6,11,"物联网基础","凌云","高等教育出版社",37,2017,5,29,"大数据处理","谭峰","北京大学出版社",37,2000,7,15,"操作系统","汤子","北京大学出版社",40.5,2004,3,15,"高级语言程序设计","张岩","吉林大学出版社",35,2010,12,18,"python程序设计","郑楚","北京大学出版社",35.5,2008,11,26};

float cb;

int nf;

scanf("%d",&nf);

cb=count(ts,20,nf);

printf("%d以后出版的书的成本为：%.2f元\n",nf,cb);

return 0;

}

26、/\*

编程：完成delsame函数，实现删除任意一个有序链表中的重复元素（相同的值保留一个）

输入数据：2 2 3 3 3 4 5 6 6 8 -1

输出数据：2 3 4 5 6 8

\*/

#include<malloc.h>

#include<stdlib.h>

#include<stdio.h>

typedef struct LNode

{

int data;

struct LNode\* next;

}LNode, \*LinkList;

void creatlist(LinkList \*head)

{

LinkList p,q;

int x;

\*head=(LinkList)malloc(sizeof(LNode));

(\*head)->next=NULL;

q=\*head;

scanf("%d",&x);

while (x!=-1)

{

p=(LinkList)malloc(sizeof(LNode));

p->data=x;

q->next=p;

q=p;

scanf("%d",&x);

}

p->next=NULL;

}

void show( LinkList L ) // 显示表

{

while(L->next)

{

printf("%d ",L->next->data);

L = L->next; }

printf("\n");

}

void delsame( LinkList L ) // 删除相同元素并释放内存

{

LinkList p,q;

p=L->next;

while (p->next!=NULL)

{

q=p->next;

if (p->data==q->data)

{p->next=q->next; free(q);}

else

p=p->next;

}

}

void main( void )

{

LinkList LA = NULL;

creatlist(&LA);

delsame(LA);

show(LA);

}

27、/\*

编程：完成search函数，实现统计数据文件zy.dat文件中的所有素数，保存在result.dat文件中

输入数据：23 25 27 29 31 78 45 789 645 641

输出数据：23 29 31 641

\*/

#include <stdio.h>

#include <stdlib.h>

int isprime(int x)

{

int i;

for(i=2;i<x;i++)

if (x%i==0) break;

if (i==x)

return 1;

else

return 0;

}

void save(char \*s)

{

FILE \*fp;

int i,x;

fp=fopen(s,"w");

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

{

scanf("%d",&x);

fprintf(fp,"%d ",x);

}

fclose(fp);

}

void search(char \*s1,char \*s2)

{

FILE \*fp1,\*fp2;

int x;

fp1=fopen(s1,"r");

fp2=fopen(s2,"w");

while ( fscanf(fp1,"%d",&x)==1)

{

if (isprime(x))

fprintf(fp2,"%d ",x);

}

fclose(fp1);

fclose(fp2);

}

void read(char \*s)

{

FILE \*fp;

int x;

fp=fopen(s,"r");

while ( fscanf(fp,"%d",&x)==1)

printf("%d ",x);

fclose(fp);

}

int main()

{

save("zy.dat");

search("zy.dat","result.dat");

read("result.dat");

printf("\n");

return 0;

}

28、/\*

编程：完成search函数，实现统计数据文件zy.dat文件中的所有成绩中不及格的成绩，保存在result.dat文件中

输入数据：65 68 89 74 71 46 91 95 59 74

输出数据：46 59

\*/

#include <stdio.h>

#include <stdlib.h>

void save(char \*s)

{

FILE \*fp;

int i,x;

fp=fopen(s,"w");

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

{

scanf("%d",&x);

fprintf(fp,"%d ",x);

}

fclose(fp);

}

void search(char \*s1,char \*s2)

{

FILE \*fp1,\*fp2;

int x;

fp1=fopen(s1,"r");

fp2=fopen(s2,"w");

while (fscanf(fp1,"%d",&x)==1)

{

if (x<60)

fprintf(fp2,"%d ",x);

}

fclose(fp1);

fclose(fp2);

}

void read(char \*s)

{

FILE \*fp;

int x;

fp=fopen(s,"r");

while (fscanf(fp,"%d",&x)==1)

printf("%d ",x);

fclose(fp);

}

int main()

{

save("zy.dat");

search("zy.dat","result.dat");

read("result.dat");

printf("\n");

return 0;

}

29、/\*

编程：针对一批学生信息（姓名，学号，成绩），完成函数ave，求学生的平均成绩。

输入数据：56 69 89 78 63 65 48 72 63 91

输出数据：平均成绩为：69.40

\*/

#include<stdio.h>

#include<stdlib.h>

#define N 10

struct Student

{

char name[20];

long num;

int score;

};

typedef struct Student student;

void save(student st[])

{

FILE \*fp;

int i;

if((fp=fopen("file4.txt","w"))==NULL)

{

printf("cannot open file\n");

return;

}

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

{

fwrite(&st[i],sizeof(student),1,fp);

}

fclose(fp);

}

float ave()

{

FILE \*fp;

int i=0;

float sum=0;

student st;

fp=fopen("file4.txt","r");

while (fread(&st,sizeof(student),1,fp)==1)

{

sum=sum+st.score;

i++;

}

fclose(fp);

return sum/i;

}

int main()

{

student stud[N]={"zy",1001,0,"sl",1002,0,"zhj",1003,0,"zzs",1004,0,"ly",1005,0,"jm",1006,0,"lf",1007,0,"zl",1008,0,"yl",1009,0,"hy",1010,0};

int i;

float s;

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

scanf("%d",&stud[i].score);

//建立数据文件

save(stud);

s=ave();

printf("平均成绩为：%.2f\n",s);

return 0;

}

30、/\*

编程：针对一批学生信息（姓名，学号，成绩），完成函数sort，评选出学生成绩的前三名。

输入数据：56 69 89 78 63 65 48 72 63 91

输出数据：zy:91,zhj:89,zzs:78

\*/

#include<stdio.h>

#include<stdlib.h>

#define N 10

struct Student

{

char name[20];

long num;

int score;

};

typedef struct Student student;

void save(student st[])

{

FILE \*fp;

int i;

if((fp=fopen("file4.txt","w"))==NULL)

{

printf("cannot open file\n");

return;

}

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

{

fwrite(&st[i],sizeof(student),1,fp);

}

fclose(fp);

}

void sort(student tt[])

{

FILE \*fp;

int i,j,k=0;

student st;

fp=fopen("file4.txt","r");

while (fread(&tt[k],sizeof(student),1,fp)==1)

k++;

fclose(fp);

for(i=0;i<k-1;i++)

for(j=i+1;j<k;j++)

if (tt[i].score<tt[j].score)

{st=tt[i];tt[i]=tt[j];tt[j]=st;}

}

int main()

{

student stud[N]={"why",1001,0,"sl",1002,0,"zhj",1003,0,"zzs",1004,0,"ly",1005,0,"jm",1006,0,"lf",1007,0,"zl",1008,0,"yl",1009,0,"zy",1010,0};

int i;

student tt[N];

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

scanf("%d",&stud[i].score);

//建立数据文件

save(stud);

sort(tt);

//输出前三名的姓名和成绩

printf("%s:%d,%s:%d,%s:%d\n",tt[0].name,tt[0].score,tt[1].name,tt[1].score,tt[2].name,tt[2].score);

return 0;

}

31、/\*

编程：现有二十张选票，三个候选人，完成fun函数，统计票王和票王得的票数。

输入数据：库里 哈登 库里 哈登 哈登 库里 阿德托昆博 库里 哈登 库里 库里 哈登 库里 哈登 哈登 库里 阿德托昆博 库里 哈登 库里

输出数据：票王是：库里,得票：10

\*/

#include <stdio.h>

#include <string.h>

#define N 20

struct person

{ char name[20];

int cnt;

};

void save()

{

FILE \*fp;

char st[20];

int i=0;

if ((fp=fopen("xp1.dat","wb"))==NULL)

{

printf("cannot open file\n");

return;

}

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

{ scanf("%s",st);

fputs(st,fp);

fputs("\r\n",fp);

}

fclose(fp);

}

struct person fun(struct person leader[])

{

FILE \*fp;

char st[20];

int i,length;

struct person pw;

fp=fopen("xp1.dat","rb");

while (fgets(st,20,fp))

{

//处理掉fgets函数读出字符串多余的换行符\r\n

length=strlen(st);

st[length-2]='\0';

for(i=0;i<3;i++)

if (strcmp(st,leader[i].name)==0) leader[i].cnt++;

}

fclose(fp);

pw=leader[0];

for(i=1;i<3;i++)

if (leader[i].cnt>pw.cnt)

pw=leader[i];

return pw;

}

int main()

{

char \*xp[N];

char data[N][20];

int i;

struct person leader[3]={"库里",0,"哈登",0,"阿德托昆博",0},pw;

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

xp[i]=data[i];

save();

pw=fun(leader);

printf("票王是：%s,得票：%d\n",pw.name,pw.cnt);

return 0;

}

32、/\*

编程：现有某高校的设备清单（设备名称，购买日期，价格），输入当前年份，完成deal函数，实现将购买时间超过6年（包括6年）的设备报废，输出新设备总值。

输入数据：2002

输出数据：新的总资产为：5150元

\*/

#include <stdio.h>

#include <string.h>

#define N 12

struct goods

{ char sbmc[30];

int gmrq;

int price;

};

void save(struct goods st[])

{

FILE \*fp;

int i=0;

if ((fp=fopen("good.dat","wb"))==NULL)

{

printf("cannot open file\n");

return;

}

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

fwrite(&st[i],sizeof(struct goods),1,fp);

fclose(fp);

}

void deal(int year)

{

FILE \*fp;

struct goods st[N],pw;

int i,j=0;

fp=fopen("good.dat","rb");

while (fread(&pw,sizeof(struct goods),1,fp)==1)

{

if (year-pw.gmrq+1<6)

st[j++]=pw;

}

fclose(fp);

fp=fopen("good.dat","wb");

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

fwrite(&st[i],sizeof(struct goods),1,fp);

fclose(fp);

}

int count()

{

int sum=0;

FILE \*fp;

struct goods pw;

fp=fopen("good.dat","rb");

while (fread(&pw,sizeof(struct goods),1,fp)==1)

sum=sum+pw.price;

fclose(fp);

return sum;

}

int main()

{

int year,zzc;

struct goods devices[20]={"AA",1995,450,"BB",1997,500,"CC",2000,300,"DD",1997,280,"EE",2011,1000,"FF",2013,1200,"GG",1996,730,"ZZYY",1979,404,"SS",2000,800,"ZZ",2015,610,"TT",2016,900,"VV",2000,340};

save(devices);

scanf("%d",&year);

deal(year);

zzc=count();

printf("新的总资产为：%d元\n",zzc);

return 0;

}

33、/\*

编程：完成search函数，实现统计数据文件zy.dat文件中的所有回文数（及逆序后和原数相等的整数，如1221），保存在result.dat文件中

输入数据：656 12321 121 7 71 46 91 95 1221 232 1979 90404 7910 1204 69898

输出数据：656 12321 121 7 1221 232

\*/

#include <stdio.h>

#include <stdlib.h>

void save(char \*s)

{

FILE \*fp;

int i,x;

fp=fopen(s,"w");

for(i=0;i<15;i++)

{

scanf("%d",&x);

fprintf(fp,"%d ",x);

}

fclose(fp);

}

void search(char \*s1,char \*s2)

{

FILE \*fp1,\*fp2;

int x,sum,t,p;

fp1=fopen(s1,"r");

fp2=fopen(s2,"w");

while (fscanf(fp1,"%d",&x)==1)

{

//判断x是否回文数

t=x;

sum=0;

while (t!=0)

{

sum=sum\*10+t%10;

t=t/10;

}

if (sum==x)

fprintf(fp2,"%d ",x);

}

fclose(fp1);

fclose(fp2);

}

void read(char \*s)

{

FILE \*fp;

int x;

fp=fopen(s,"r");

while (fscanf(fp,"%d",&x)==1)

printf("%d ",x);

fclose(fp);

}

int main()

{

save("zy.dat");

search("zy.dat","result.dat");

read("result.dat");

printf("\n");

return 0;

}

34、/\*

编程：一批学生成绩信息（姓名，成绩1，成绩2，成绩3，成绩4），完成search函数，统计本次考试中某道题(1~4)的得分率。

输入数据：2

输出数据：第2题得分率：73.6%

\*/

#include <stdio.h>

#include <stdlib.h>

struct Student

{

char name[20];

int score[4];//依次存储该生每一道题的得分，原始题目分值都是25分

};

typedef struct Student student;

void save(char \*s,student stu[])

{

FILE \*fp;

int i,x;

fp=fopen(s,"w");

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

fwrite(&stu[i],sizeof(student),1,fp);

fclose(fp);

}

void search(char \*s1,char \*s2)

{

FILE \*fp1,\*fp2;

student st;

int i=0;

float sum1,sum2,sum3,sum4;

fp1=fopen(s1,"r");

fp2=fopen(s2,"w");

sum1=sum2=sum3=sum4=0;

while (fread(&st,sizeof(student),1,fp1)==1)

{

sum1=sum1+st.score[0];

sum2=sum2+st.score[1];

sum3=sum3+st.score[2];

sum4=sum4+st.score[3];

i++;

}

sum1=sum1/(i\*25);sum2=sum2/(i\*25);sum3=sum3/(i\*25);sum4=sum4/(i\*25);

fprintf(fp2,"%f %f %f %f ",sum1,sum2,sum3,sum4);

fclose(fp1);

fclose(fp2);

}

void read(char \*s,int n)

{

FILE \*fp;

float x;

int i=0;

fp=fopen(s,"r");

while (fscanf(fp,"%f",&x)==1)

{

i++;

if (i==n)

{printf("第%d题得分率：%.1f%%",i,x\*100);break;}

}

fclose(fp);

}

int main()

{

student stu[]={"AA",18,17,25,20,"BB",20,10,15,25,"CC",20,15,20,12,"DD",18,20,20,10,"EE",16,18,20,18,"FF",10,25,20,10,"GG",10,25,20,10,"HH",20,19,20,20,"II",20,10,25,20,"JJ",20,25,20,16};

int n;

save("zy.dat",stu);

search("zy.dat","result.dat");

printf("输入题目编号：");

scanf("%d",&n);

read("result.dat",n);

printf("\n");

return 0;

}