#include <stdio.h>  
#include <stdlib.h>  
struct node  
{  
int a,b,c,d;  
};  
int main()  
{  
//以二进制方式读取问件  
FILE \*fp=fopen("C:\\1.bin","rb");  
//把fp移到文件末尾  
fseek(fp,0L,2);  
//得到文件的字节数  
int i=ftell(fp);  
printf("字节数：%d\n",i);  
//把fp移到文件开头  
fseek(fp,0L,0);  
//得到文件中一共有多少个结构体  
int count=i/(sizeof(struct node));  
printf("结构体个数：%d\n",count);  
//动态申请一段内存空间，并让myNode指向空间的起始位置  
struct node\* myNode=(struct node \*)malloc(sizeof(struct node)\*count);  
//读取所有数据  
fread(myNode,sizeof(struct node),count,fp);  
//关闭文件  
fclose(fp);  
//输出前10组数据  
printf("前10组数据:\n");  
for(int j=0;j<10;j++)  
{  
printf("%6d%6d%6d%6d\n",myNode[j].a,myNode[j].b,myNode[j].c,myNode[j].d);  
}  
//释放内存  
free(myNode);  
  
}

work 4

// 临时.cpp : 定义控制台应用程序的入口点。

//

#include \"stdafx.h\"

int \_tmain(int argc, \_TCHAR\* argv[])

{

int num,N,num1,num2,num3,num4,num5;

printf(\"请输入一个不超过五位的整数：\");

scanf(\"%d\",&num);

if(num>99999 || num<1)

printf(\"输入无效\");

else {if(num>9999)

N=5;

else {if(num>999)

N=4;

else {if(num>99)

N=3;

else {if(num>9)

N=2;

else

N=1;

}

}

}

}

num1=num/10000;

num2=(num%10000)/1000;

num3=(num%1000)/100;

num4=(num%100)/10;

num5=num%10;

printf(\"它是%d 位数\",N);

printf(\"\\n每位数为：%d, %d, %d, %d, %d \\n\",num1,num2,num3,num4,num5);

if(N==5)

printf(\"逆序数为：%d%d%d%d%d\",num5,num4,num3,num2,num1);

if(N==4)

printf(\"逆序数为：%d%d%d%d\",num5,num4,num3,num2);

if(N==3)

printf(\"逆序数为：%d%d%d\",num5,num4,num3);

if(N==2)

printf(\"逆序数为：%d%d\",num5,num4);

if(N==1)

printf(\"逆序数为：%d\",num5);

return 0;

}

作业标题：03.switch与简单循环

1.用户输入年和月，用switch语句实现显示该月有多少天

2.编写程序打印如下图形共五行

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

3.编写程序让用户输入月日，计算该天是该年的第多少天。(不考虑闰年)

Work 1

// 月份问题.cpp : 定义控制台应用程序的入口点。

//

#include \"stdafx.h\"

int \_tmain(int argc, \_TCHAR\* argv[])

{

int year,month;

printf(\"请输入年份，月份:\\n\");

scanf(\"%d,%d\",&year,&month);

if(month!=2)

{

switch(month)

{

case 1:

case 3;

case 5:

case 7:

case 8:

case 10:

case 12:printf(\"该月有31天\\n\");break;

case 4:

case 6:

case 9:

case 11:printf(\"该月有30天\\n\");break;

}

}

else

{

if (year%400==0 || (year%4==0 && year%100!=0))

printf(\"该月有29天\\n\");

else

printf(\"该月有28天\\n\");

}

return 0;

}

Work 2

// 5行星星.cpp : 定义控制台应用程序的入口点。

//

#include \"stdafx.h\"

int \_tmain(int argc, \_TCHAR\* argv[])

{

int i=1,j,g;

while (i<=5)

{

j=1;

while(j<=(5-i))

{

printf(\" \");

j++;

}

g=1;

while(g<=i)

{

printf(\"\*\");

g++;

}

i++;

printf(\"\\n\");

}

return 0;

}

Work 3

// 日数问题.cpp : 定义控制台应用程序的入口点。

//

#include \"stdafx.h\"

int \_tmain(int argc, \_TCHAR\* argv[])

{

int month,day,N;

printf(\"请输入月，日：\\n\");

scanf(\"%d,%d\",&month,&day);

switch(month)

{

case 1:N=day;break;

case 2:N=31+day;break;

case 3:N=31+28+day;break;

case 4:N=31\*2+28+day;break;

case 5:N=31\*2+28+30+day;break;

case 6:N=31\*3+28+30+day;break;

case 7:N=31\*3+28+30\*2+day;break;

case 8:N=31\*4+28+30+2+day;break;

case 9:N=31\*5+28+30\*2+day;break;

case 10:N=31\*5+28+30\*3+day;break;

case 11:N=31\*6+28+30\*3+day;break;

case 12:N=31\*6+28+30\*4+day;break;

}

printf(\"%d月%d日是第%d 天\\n\",month,day,N);

return 0;

}

作业标题：07.循环与数组

1。.编写程序对10个整数进行排序（从小到大，选择排序）

2.编写程序将整型数组中10个元素先后顺序颠倒存储

3.编写程序计算一个3\*3矩阵每行 每列 和对角线的和

4。公园前5世纪，我国数学家张丘建在《算经》中提出百钱百鸡问题：鸡翁一值钱五，鸡母一值钱三，鸡雏三值钱一，百钱买百鸡，问鸡翁、母、雏各几何？

附加题

1.已知2010年1月1日是星期五,编写程序计算用户输入的一个日期的星期号

作业内容： （附件：无）

// 排序.cpp : 定义控制台应用程序的入口点。

//

#include \"stdafx.h\"

#include <stdlib.h>

int \_tmain(int argc, \_TCHAR\* argv[])

{

int num[10] ;

int i,j,t;

for(i=0;i<=9;i++)

{

num[i]=rand();

printf(\"%d\\n\",num[i]);

}

for(i=0;i<=9;i++)

{

for(j=0;j<=9;j++)

{

if (num[i]<num[j])

{

t=num[i];

num[i]=num[j];

num[j]=t;

}

}

}

printf(\"排序后数列为:\\n\");

for(i=0;i<=9;i++)

{

printf(\"%d\\n\",num[i]);

}

return 0;

}

// 颠倒.cpp : 定义控制台应用程序的入口点。

//

#include \"stdafx.h\"

#include <stdlib.h>

int \_tmain(int argc, \_TCHAR\* argv[])

{

int num[10];

int i,j;

for(i=0;i<=9;i++)

{ num[i]=rand();

printf(\"%d\\n\",num[i]);

}

printf(\"颠倒后排列顺序为：\\n\");

for(i=9;i>=0;i--)

printf(\"%d\\n\",num[i]);

return 0;

}

// 矩阵.cpp : 定义控制台应用程序的入口点。

//

#include \"stdafx.h\"

#include <stdlib.h>

int \_tmain(int argc, \_TCHAR\* argv[])

{

int num[3][3];

int i,j;

int sumline[3]={},sumrow[3]={},sum=0;

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

{

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

{

num[i][j]=rand();

printf(\"%8d\",num[i][j]);

}

printf(\"\\n\");

}

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

{

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

{

sumline[i]=sumline[i]+num[i][j];

}

printf(\"\\n第%d行的和为：%d\\n\",i+1,sumline[i]);

}

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

{

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

{

sumrow[j]=sumrow[j]+num[i][j];

}

printf(\"\\n第%d列的和为：%d\\n\",j+1,sumrow[j]);

}

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

sum=sum+num[i][i];

printf(\"\\n对角线和为%d：\\n\",sum);

return 0;

}

// 百鸡问题.cpp : 定义控制台应用程序的入口点。

//

#include \"stdafx.h\"

int \_tmain(int argc, \_TCHAR\* argv[])

{

int cock,hen,chicken;

for(cock=0;cock<=20;cock++)

{

for(hen=0;hen<=33;hen++)

{

chicken=(100-cock-hen);

if(100==(5\*cock+3\*hen+chicken/3.0))

printf(\"百钱买百鸡，公鸡%d只，母鸡%d只，鸡雏%d只。\\n\",cock,hen,chicken);

}

}

return 0;

}

// 指针回文.cpp : 定义控制台应用程序的入口点。

//

#include "stdafx.h"

int judge(char \*src1,char \*src2);

void change(char \*pointer);

/\*=========================================

本程序用于判断字符串是否是回文

字符数组：str

用于判断结果的整形变量：result：1：是回文串；0：不是回文串；

指针变量：\*p1：从字符串开头检索；\*p2：从字符串结尾检索

============================================\*/

int \_tmain(int argc, \_TCHAR\* argv[])

{

char str[20];

int result;

char \*p1,\*p2;

printf("请输入一个字符串：\n");

gets(str);

p1=str;

p2=str;

while(\*p2!='\0')

p2++;

p2--;

result=judge(p1,p2);

if(result==1)

printf("是回文串\n");

else

printf("不是回文串\n");

return 0;

}

/\*=========================================

本函数用于判断字符串是否是回文

形式参数：\*src1,\*src2;

实际参数：p1,p2;

返回值：是回文串：1；

不是回文串：0；

============================================\*/

int judge(char \*src1,char \*src2)

{

while(src1<src2)

{

while(\*src1<65 || (\*src1>90 && \*src1<97) || \*src1>122)

{

src1++;

}

while(\*src2<65 || (\*src2>90 && \*src2<97) || \*src2>122)

{

src2--;

}

change(src1);

change(src2);

if(\*src1==\*src2)

{

src1++;

src2--;

}

else

break;

}

if(src1>=src2)

return 1;

else

return 0;

}

/\*=========================================

本函数用于对字符大小写进行转换统一

形式参数：pointer

实际参数：src1,src2

============================================\*/

void change(char \*pointer)

{

if(\*pointer>=65 && \*pointer<=90)

\*pointer=\*pointer+32;

}

作业标题：17.结构体初步

1.设有结构体描述如下：

struct person

{

char name[10];

int sex; //0表示男生，1表示女生

int age;

};

在本次的作业的附件中存储了10个该结构体的数据，该文件是二进制文件，要求：

1）读取出这10个数据，并在屏幕上显示

2）计算出他们的平均年龄

3）删除年龄最大和最小的两位

4）将处理完的数据写入到一个文本文件中，要求，每行存一个人的信息，姓名站10列，性别显示为“男”或者“女”，年龄占5列，左对齐。

2.编写一个通讯录程序

要求

1）提供添加、删除、修改和清除的功能

2）通讯录人数不超过80人

3）通讯录保存在二进制文件中

4）程序界面友好，函数设计合理

// 结构体.cpp : 定义控制台应用程序的入口点。

//

#include \"stdafx.h\"

#include <stdio.h>

#include <string.h>

#define LEN sizeof(struct person)

void max(struct person per[]);

void min(struct person per[]);

void save();

float aver\_age=0;

struct person

{

char name[10];

int sex; //0表示男生，1表示女生

int age;

};

struct person per[10];

/\*=======================================

本函数用于对一组学生数据进行处理

目标文件：1947.dat 以二进制形式打开

========================================\*/

int \_tmain(int argc, \_TCHAR\* argv[])

{

int i;

FILE \*fp;

fp=fopen(\"d:\\\\1947.dat\",\"rb\");

if(fp==NULL)

{

printf(\"can not open!\\n\");

return 0;

}

for(i=0;i<=9;i++)

{

fread(&per[i],LEN,1,fp);

printf(\"%10s%4d%4d\\n\",per[i].name,per[i].sex,per[i].age);

}

fclose(fp);

for(i=0;i<=9;i++)

{

aver\_age=aver\_age+per[i].age;

}

aver\_age=aver\_age/10.0;

max(per);

min(per);

save();

return 0;

}

/\*=======================================

本函数用于对数据中最大年龄者数据进行删除

变量：i； num：确定最值数据

========================================\*/

void max(struct person per[])

{

int i;

int num=0;

int temp=per[0].age;

for(i=0;i<=9;i++)

{

if(temp<per[i].age)

{

temp=per[i].age;

num=i;

}

}

for(i=num;i<=8;i++)

{

per[i]=per[i+1];

}

}

/\*=======================================

本函数用于对数据中最小年龄者数据进行删除

变量：i； num：确定最值数据

========================================\*/

void min(struct person per[])

{

int i;

int num=0;

int temp=per[0].age;

for(i=0;i<=8;i++)

{

if(temp>per[i].age)

{

temp=per[i].age;

num=i;

}

}

for(i=num;i<=7;i++)

{

per[i]=per[i+1];

}

}

/\*=======================================

本函数用于对数据结果进行保存

目的文件：d:\\\\123.txt

========================================\*/

void save()

{

FILE \*fp2;

int i;

fp2=fopen(\"d:\\\\123.txt\",\"w+\");

if(fp2==NULL)

{

printf(\"can not open file\");

return;

}

fprintf(fp2,\"平均年龄为 %f\\n\\n\\n\",aver\_age);

for(i=0;i<=7;i++)

{

if(per[i].sex==1)

fprintf(fp2,\"%-10s 女 %-5d\\n\",per[i].name,per[i].age);

else

fprintf(fp2,\"%-10s 男 %-5d\\n\",per[i].name,per[i].age);

}

fclose(fp2);

}

// 电话簿.cpp : 定义控制台应用程序的入口点。

//

#include "stdafx.h"

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

struct person

{

char name[10];

char tel[20];

char add[50];

struct person \*next;

};

struct person \*head=NULL;

#define LEN sizeof(struct person)

struct person \*open();//打开并读入

void print(struct person \*head);

void reserve(struct person \*head);//保存修改

int menu();//菜单

struct person \*insert(struct person \*head);//添加

struct person \*dele(struct person \*head);//删除

struct person \*revise(struct person \*head);//修改

struct person \*remove(struct person \*head);//清除

int \_tmain(int argc, \_TCHAR\* argv[])

{

int chose;

do{

head=open();

print(head);

chose=menu();

switch (chose)

{

case 1:head=insert(head);break;

case 2:head=dele(head);break;

case 3:head=revise(head);break;

case 4:head=remove(head);break;

case 5:return 0;

}

reserve(head);

}while(1);

}

/\*=========================================

读取源通讯录

===========================================\*/

struct person \*open()

{

int n=0;

struct person \*p1=NULL;

struct person \*p2=NULL;

FILE \*fp;

fp=fopen("d:\\123.txt","r+");

if(fp==NULL)

{

printf("无法打开源文件\n");

}

else

{

p1=p2=(struct person \*)malloc(LEN);

fscanf(fp,"%s %s %s",p1->name,p1->tel,p1->add);

while(!feof(fp))

{

n=n+1;

if(n==1)

head=p1;

else p2->next=p1;

p2=p1;

p1=(struct person \*)malloc(LEN);

fscanf(fp,"%s %s %s",p1->name,p1->tel,p1->add);

}

p2->next=NULL;

}

fclose(fp);

return (head);

}

void print(struct person \*head)

{

struct person \*p1;

p1=head;

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

printf(" 电 话 簿 \n");

printf("%-10s%-20s%-50s","姓名","电话","地址\n");

while(p1->name!=NULL)

{

printf("%10s%20s%50s",p1->name,p1->tel,p1->add);

p1=p1->next;

}

}

/\*==========================================

显示操作菜单

==========================================\*/

int menu()

{ int chose;

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

printf(" 编辑联系人\n");

printf(" 1：添加\n");

printf(" 2：删除\n");

printf(" 3：修改\n");

printf(" 4：清除\n");

printf(" 5：退出\n");

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

do

scanf("%d",&chose);

while(chose!=1 && chose!=2 && chose!=3 && chose!=4 && chose!=5);

return (chose);

}

/\*======================================

保存更改

========================================\*/

void reserve(struct person \*head)

{

struct person \*p1;

p1=head;

FILE \*fp;

fp=fopen("d:\\123.txt","w");

while(p1!=NULL)

{

fprintf(fp,"%s %s %s ",p1->name,p1->tel,p1->add);

p1=p1->next;

}

fclose(fp);

}

/\*==========================================

添加联系人

==========================================\*/

struct person \*insert(struct person \*head)

{

struct person \*ins=NULL;

struct person \*p1=NULL,\*p2=NULL;

printf("请输入：\n姓名：\n");

ins=(struct person \*)malloc(LEN);

scanf("%s",ins->name);

printf("电话：\n");

scanf("%s",ins->tel);

printf("地址：\n");

scanf("%s",ins->add);

p1=head;

if(head==NULL)

{

head=ins;

ins->next=NULL;

}

else

{

while(p1->next!=NULL)

{

p2=p1;

p1=p1->next;

}

p1->next=ins;

ins->next=NULL;

}

printf("完成！已添加 %s 到您的电话簿中！\n\n",ins->name);

return (head);

}

/\*=========================================

删除联系人

==========================================\*/

struct person \*dele(struct person \*head)

{

struct person \*p1,\*p2;

char tname[10];

if(head==NULL)

{

printf("电话簿为空！\n");

return (head);

}

printf("请输入删除联系人的姓名：\n");

scanf("%s",tname);

p1=head;

while(strcmp(tname,p1->name)!=0 && p1->next!=NULL)

{

p2=p1;

p1=p1->next;

}

if(strcmp(tname,p1->name)==0)

{

if(p1==head)

head=p1->next;

else

p2->next=p1->next;

printf("完成！已从您的电话簿中删除：%s\n\n",tname);

}

else

printf("没有该联系人！\n");

return (head);

}

/\*=========================================

修改联系人

==========================================\*/

struct person \*revise(struct person \*head)

{

char tname[10];

printf("请输入修改联系人的名字\n");

scanf("%s",tname);

struct person \*p1;

p1=head;

while(strcmp(p1->name,tname)!=0 && p1->next!=NULL)

{

p1=p1->next;

}

if(strcmp(p1->name,tname)!=0)

printf("没有该联系人\n");

else

{

printf("请重新输入：\n");

scanf("%s",p1->name);

scanf("%s",p1->tel);

scanf("%s",p1->add);

}

return (head);

}

/\*=========================================

清除联系人

==========================================\*/

struct person \*remove(struct person \*head)

{

head=NULL;

return (head);

}

作业标题：18.链表初步

1.设有结构体描述如下：

struct person

{

char name[10];

int sex; //0表示男生，1表示女生

int age;

};

在本次的作业的附件中存储了多个该结构体的数据，具体数量未知，该文件是二进制文件，要求：

1）读取出这10个数据，并在屏幕上显示

2）计算出他们的平均年龄

3）删除年龄大于60的人员

4）将处理完的数据写入到一个文本文件中，要求，女生在前，男生在后，每行存一个人的信息，姓名占10列，左对齐，性别显示为“男”或者“女”，年龄占5列，右对齐。

// 链表初步.cpp : 定义控制台应用程序的入口点。

//

#include "stdafx.h"

#include <stdio.h>

#include <malloc.h>

void print(struct person \*head);

struct person \*open();

float aver(struct person \*head);

struct person \*dele(struct person \*head);

void reserve(struct person \*head,float average);

#define LEN sizeof(struct person1)

/\*=====================================

本结构体用于计算读取数据长度

====================================\*/

struct person1

{

char name[10];

int sex; //0表示男生，1表示女生

int age;

};

struct person

{

char name[10];

int sex; //0表示男生，1表示女生

int age;

struct person \*next;

};

/\*=====================================

本程序用于读取附件中存储的多个结构体的数据；

并对该数据进行处理，结果保存在“d:\\数据结果.txt”中；

====================================\*/

int \_tmain(int argc, \_TCHAR\* argv[])

{

float average;

struct person \*head=NULL;

head=open();

print(head);

average=aver(head);

printf("平均年龄为：%f\n",average);

head=dele(head);

print(head);

reserve(head,average);

return 0;

}

/\*===================================================

本函数用于读取"d:\\1951.dat"文件，并将数据储存在链表中

形式参数：struct person \*p1,\*p2；

实际参数：struct person \*head；

=====================================================\*/

struct person \*open()

{

int n=0;

struct person \*head=NULL;

struct person \*p1,\*p2;

p1=head;

FILE \* fp;

fp=fopen("d:\\1951.dat","rb");

if(fp==NULL)

{

printf("读取失败！\n");

return (head);

}

p1=p2=(struct person \*)malloc(LEN);

fread(p1,LEN,1,fp);

while(!feof(fp))

{

n=n+1;

if(n==1) head=p1;

else p2->next=p1;

p2=p1;

p1=(struct person \*)malloc(LEN);

fread(p1,LEN,1,fp);

}

p2->next=NULL;

fclose(fp);

return(head);

}

/\*=====================================

本程序用于将储存在链表中的数据显示在屏幕上

形式参数：struct person \*p1；

实际参数：struct person \*head；

====================================\*/

void print(struct person \*head)

{

struct person \*p1;

p1=head;

while(p1!=NULL)

{

printf("%10s%4d%4d\n",p1->name,p1->sex,p1->age);

p1=p1->next;

}

}

/\*=====================================

本函数用于求数据中年龄的平均值

参数：struct person \*p1；float average；

传入待计算：struct person \*head；

====================================\*/

float aver(struct person \*head)

{

struct person \*p1;

float average=0;

int count=0;

p1=head;

while(p1!=NULL)

{

average=average+p1->age;

p1=p1->next;

count++;

}

average=average/count;

return (average);

}

/\*=====================================

本函数用于删除数据中年龄大于60的个体数据；

参数：struct person \*p1,\*p2

传入待计算：struct person \*head；

====================================\*/

struct person \*dele(struct person \*head)

{

struct person \*p1,\*p2;

p1=head;

p2=p1;

while(p1!=NULL)

{

if(p1->age>60)

{

if(p1==head)

{

p1=p1->next;

head=p1;

p2=p1;

}

else

{

p2->next=p1->next;

p1=p1->next;

}

}

else

{

p2=p1;

p1=p1->next;

}

}

return (head);

}

/\*=====================================

本函数用于将处理后的数据结果保存

参数：struct person \*p1;

传入待计算：struct person \*head；

保存路径：d:\\数据结果.txt；

====================================\*/

void reserve(struct person \*head,float average)

{

struct person \*p1;

FILE \*fp;

fp=fopen("d:\\数据结果.txt","w+");

p1=head;

fprintf(fp,"平均年龄为：%f\n",average);

while(p1!=NULL)

{

if(p1->sex==1)

fprintf(fp,"%-10s女%5d\n",p1->name,p1->age);

p1=p1->next;

}

p1=head;

while(p1!=NULL)

{

if(p1->sex==0)

fprintf(fp,"%-10s男%5d\n",p1->name,p1->age);

p1=p1->next;

}

fclose(fp);

}