**哈尔滨工程大学**

**《计算思维二》实验报告**

**基础实践一**

姓 名： 吉旭洋 班 级： 20201081

学 号： 2020108104

实验时间: 2021 年 5 月 8日

|  |  |
| --- | --- |
| 成绩 |  |

**哈尔滨工程大学计算机基础课程教学中心**

实验题目一1：由键盘输入两个整数，分别计算这两个数的和，差，积，商。

设计思想：通过定义整型变量，利用scanf函数进行数字赋值计算，同时利用printf函数进行输出。

实验代码及注释：#include<stdio.h>

int main()

{

int a,b,s,c,f,j;

scanf("%d%d",&a,&b);

s=a+b;

c=a-b;

j=a\*b;

f=a/b;

printf("和=%d\n",s);

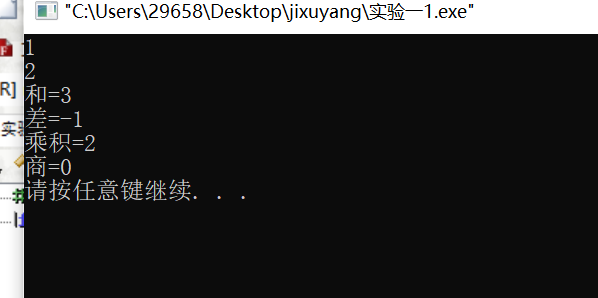
printf("差=%d\n",c);

printf("乘积=%d\n",j);

printf("商=%d\n",f);

return 0;

}

验证与结论：

总结与心得体会：可以简便的计算一对整数的和差积商。很好的学习了整型变量和scanf的应用。

实验题目一2：要将“China”译成密码，译码规律是：用原来字母后面的第4个字母代替原来的字母。例如，字母A后面第4个字母是E。E代替A。因此，“China”应译为“Glmre”。请编一程序，用赋初值的方法使c1、c2、c3、c4、c5五个变量的值分别为‘C’、‘h’、‘i’、‘n’、‘a’，经过运算，使c1、c2、c3、c4、c5分别变成为‘G’、‘l’、‘m’、‘r’、‘e’，并输出。

设计思想：输入函数，利用ASCII码转换，将字母转换为数字，进行运算找到对应的字符，达到加密的效果。

实验代码及注释：#include<stdio.h>

int main()

{

char c1='C',c2='h',c3='i',c4='n',c5='a';

c1='C'+4;

c2='h'+4;

c3='i'+4;

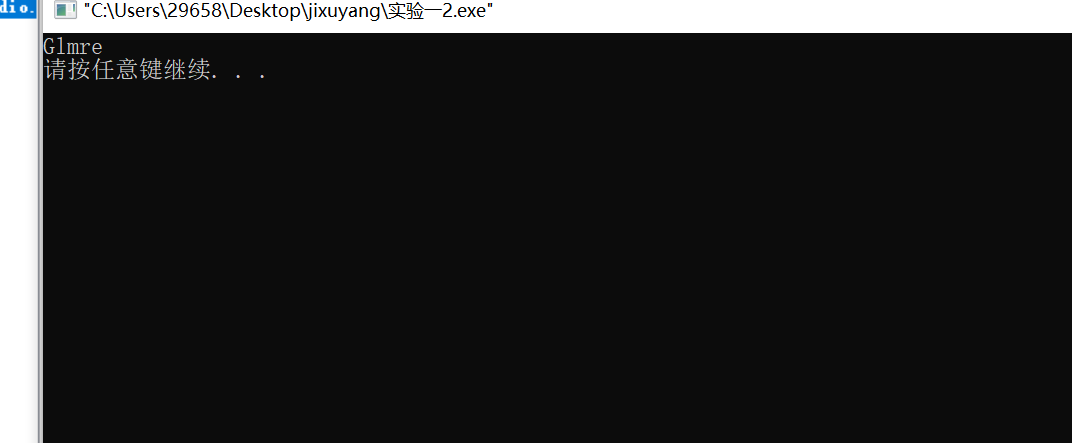
c4='n'+4;

c5='a'+4;

printf("%c%c%c%c%c\n",c1,c2,c3,c4,c5);

return 0;

}

验证与结论：

可以输出题目要求的字符转换。

总结与心得体会：

ASCII码十分重要，学好之后可以进行文字加密进行传输。

实验题目一3： 设有变量定义如下：

Int i=6,j=12;

Double x=3.28,y=90;

希望得到如下输出结果：

i=6 j=c

x=3.280000E+000 y=90

请编程实现

设计思想：定义整型变量与字符进行替换，最后由printf函数输出。

实验代码及注释：#include<stdio.h>

int main()

{

int i=6,j=12;

double x=3.28,y=90;

j='c';

y=90;

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

printf("j=%c\n",j);

printf("x=%fE+000\n",x);

printf("y=%.0f\n",y);

return 0;

}

验证与结论：



总结与心得体会：学习了浮点数保留小数位数，以及数字与字符的转换。

实验题目一4：某种物品每年的折扣费的线性计算方法如下：折扣费=（购买价格—废品价值）/使用年限。请编写一个程序，当输入某物品的购买价格、使用年限和废品价值时，程序能计算出其在某一年折旧后的价值（结果保留两位小数）。

设计思想：

通过float以及scanf函数，利用题中所给的函数进行运算，再通过%.2f函数保留两位小数，从而达到计算效果。

实验代码及注释：#include<stdio.h>

int main()

{

int price=0,year=0,value=0,cost=0;

scanf("%d%d%d",&price,&year,&value);

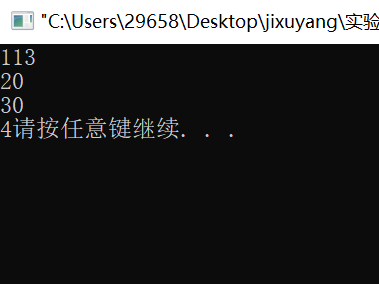
cost=(price-value)/year;

printf("%d",cost);

return 0;

}

验证与结论：



经过多组数据的输入发现，该程序确可以输出正确数值，从而达到解决实际问题的作用。

总结与心得体会：可以很好的计算最后商品的价值。省了人为复杂的计算。更加熟练的掌握了scanf函数和printf。

实验题目一5：编写程序实现以下功能。计算在贷款第一个月，第二个月及第三个月后需要还款的金额。贷款金额：20000.00 年贷款利率：6.0% 每个月还款金额：386.66 第一个月剩余的需还款金额：19713.34 第二个月剩余的需还款金额：19425.25 第三个月剩余的需还款金额：19135.71 说明：所有数额有效数字保持在小数点后两位。提示：每个月，剩余的贷款金额为总数减去每个月的还款金额，但是每个月剩余的贷款金额要加上按照月贷款利率计算出来的利息。月贷款利率为年贷款利率除以12。

设计思想：先分出三种范围，再分别进行运算

实验代码及注释：#include<stdio.h>

int main()

{

float loan=20000.00,rate=0.005,repayment=386.66,l1,l2,l3;//定义浮点数

l1=loan-repayment;

l1=l1\*(1+rate);

l2=l1-repayment;

l2=l2\*(1+rate);

l3=l2-repayment;

l3=l3\*(1+rate);

printf("%.2f\n",l1);//输出小数

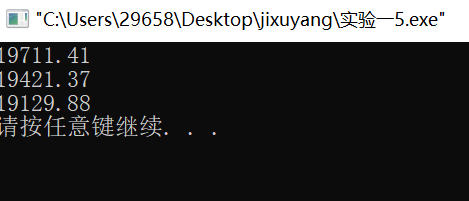
printf("%.2f\n",l2); //输出小数

printf("%.2f\n",l3); //输出小数

return 0;

}

验证与结论：



得到了题目需要的金额。

总结与心得体会：更好的掌握了浮点数，输出小数。

实验题目二1：编写程序，计算购货款。请输入购货金额，输出实际付款金额。

购货折扣率如下：

购货金额<=500元 不打折

500元<购货金额<=1000元 9折

1000元<购货金额 8折

分别用if语句、switch语句编程。

设计思想：先分出三种范围，再分别进行运算

实验代码及注释：《1》if选择结构

#include<stdio.h>

int main()

{

float r1=0.9,r2=0.8;

float price,cost;

scanf("%f",&price);

if(0<price&&price<=500)

cost=price;

else if(500<price&&price<=1000)

cost=0.9\*price;

else if(price>1000)

cost=0.8\*price;

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

return 0;

}

《2》switch结构：

#include<stdio.h>

int main()

{

int cost,price,cost1,cost2,cost3;

printf("please enter the purchase price\n");

scanf("%d",&price);

cost1=price;

cost2=0.9\*price;

cost3=0.8\*price;

switch(price/100)

{

case 0:printf("cost=%d\n",cost1);break;

case 1:printf("cost=%d\n",cost1);break;

case 2:printf("cost=%d\n",cost1);break;

case 3:printf("cost=%d\n",cost1);break;

case 4:printf("cost=%d\n",cost1);break;

case 5:printf("cost=%d\n",cost2);break;

case 6:printf("cost=%d\n",cost2);break;

case 7:printf("cost=%d\n",cost2);break;

case 8:printf("cost=%d\n",cost2);break;

case 9:printf("cost=%d\n",cost2);break;

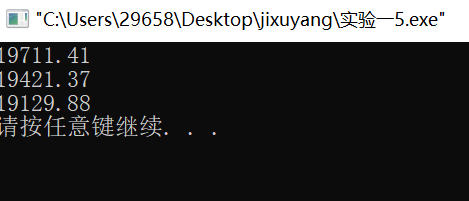
default:printf("cost=%d\n",cost3);

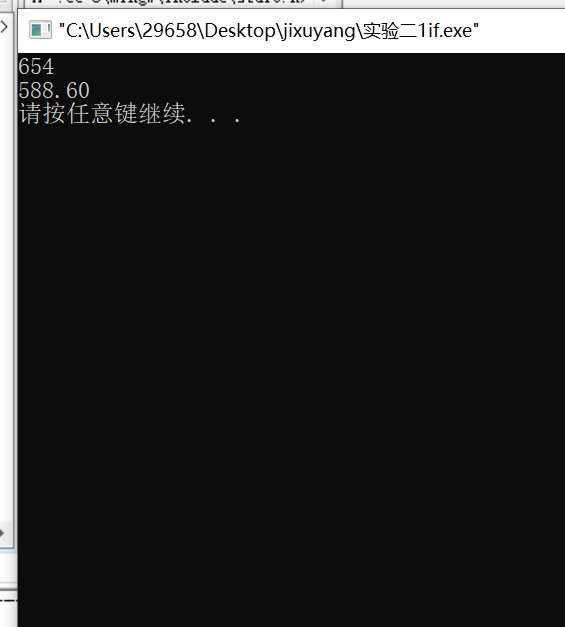
}

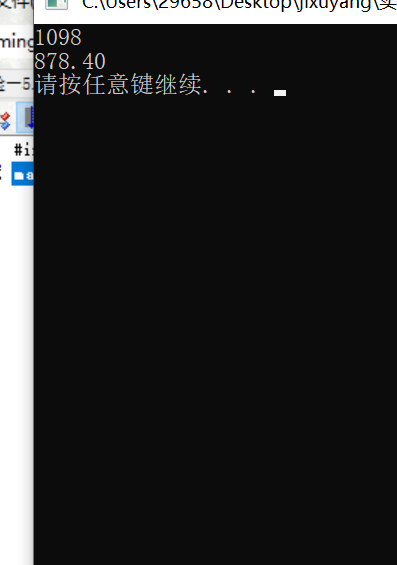
return 0;

}

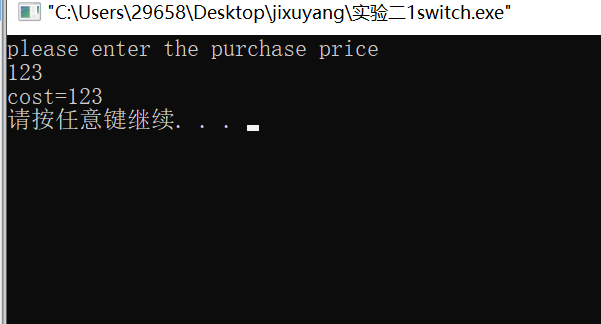
验证与结论：《1》if：



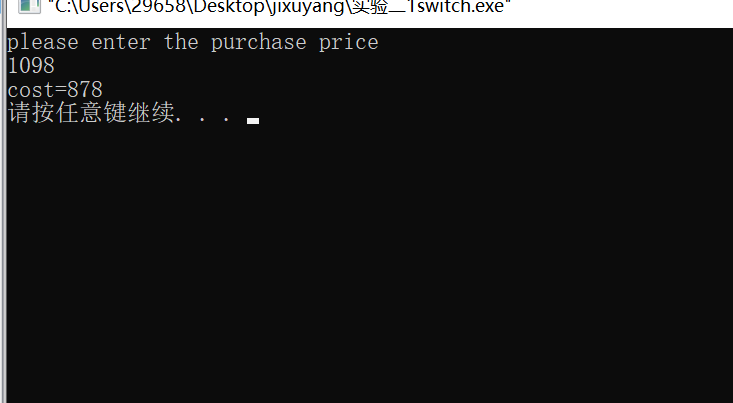




《2》switch：







二者均可以计算出不同用电下用户需要交的电费。

总结与心得体会：在if选择结构下本题中利用else if的语句很重要，若全用if语句将出现同一个数符合两个范围，在两个范围中同时进行运算情况此外在switch语句中case的运用也尤为重要，break需要跳出该case判断。

实验题目二2：输入4个整数，求其前两个数中较小的数与后两个数中较大的数之和。

设计思想：利用与或运算，借用if语句。先比较大小，再进行计算

实验代码及注释：#include<stdio.h>

int main()

{

int a,b,c,d,sum;

int max,min;

scanf("%d%d%d%d",&a,&b,&c,&d);

if(a<b)//判断语句

max=b;//选择语句

else

max=a;

if(c<d)

min=c;

else

{min=d;}

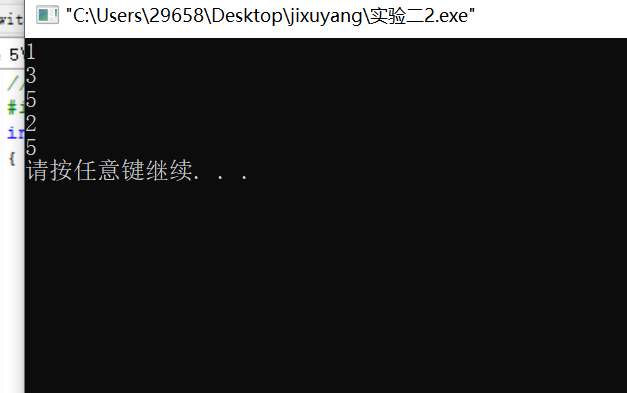
sum=max+min;

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

return 0;

}

验证与结论：



经过输入多组数据，验证得出，该程序确可以进行程序运算，计算出结果，做到筛选求和效果。

总结与心得体会：此程序输入为整数，我认为可以定义双精度数使此程序更加广泛的应用。同时练习了与，或运算的符号，注意为“||”或者“&&”，两个一起使用。

实验题目二3： 输入8个整数，求其中所有偶数的和。

设计思想：通过if语句判断是否为偶数，最终进行求和。

实验代码及注释：#include<stdio.h>

int main()

{

int a1,a2,a3,a4,a5,a6,a7,a8,sum=0;//定义整型变量

scanf("%d%d%d%d%d%d%d%d",&a1,&a2,&a3,&a4,&a5,&a6,&a7,&a8);

if(a1>0&&a1%2==0)//判断是否为偶数

sum=sum+a1;

else

sum=sum+0;//求和

if(a2>0&&a2%2==0)

sum=sum+a2;

else

sum=sum+0;

if(a3>0&&a3%2==0)

sum=sum+a3;

else

sum=sum+0;

if(a4>0&&a4%2==0)

sum=sum+a4;

else

sum=sum+0;

if(a5>0&&a5%2==0)

sum=sum+a5;

else

sum=sum+0;

if(a6>0&&a6%2==0)

sum=sum+a6;

else

sum=sum+0;

if(a7>0&&a7%2==0)

sum=sum+a7;

else

sum=sum+0;

if(a8>0&&a8%2==0)

sum=sum+a8;

else

sum=sum+0;

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

return 0;

}

验证与结论：

通过if-else语句判断出数字是否为偶数从而进行偶数求和的到结果。

总结与心得体会：利用选择结构判断偶数，在&还有||的理解上更上一层楼。

实验题目二4： 输入10个整数，输出其中的最大值和最小值。

设计思想：利用scanf函数进行数字的输入，利用数组对数字进行减少代码行数，再利用循环结构判断出max与min。

实验代码及注释：#include<stdio.h>

int main(void)

{

int i,max=-888,min=1000;

int a[10];

printf("please input 10 digits\n");

for(i=0;i<10;i++)//循环结构，十次判断

{

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

if(a[i]>max)max=a[i];

if(a[i]<min)min=a[i];

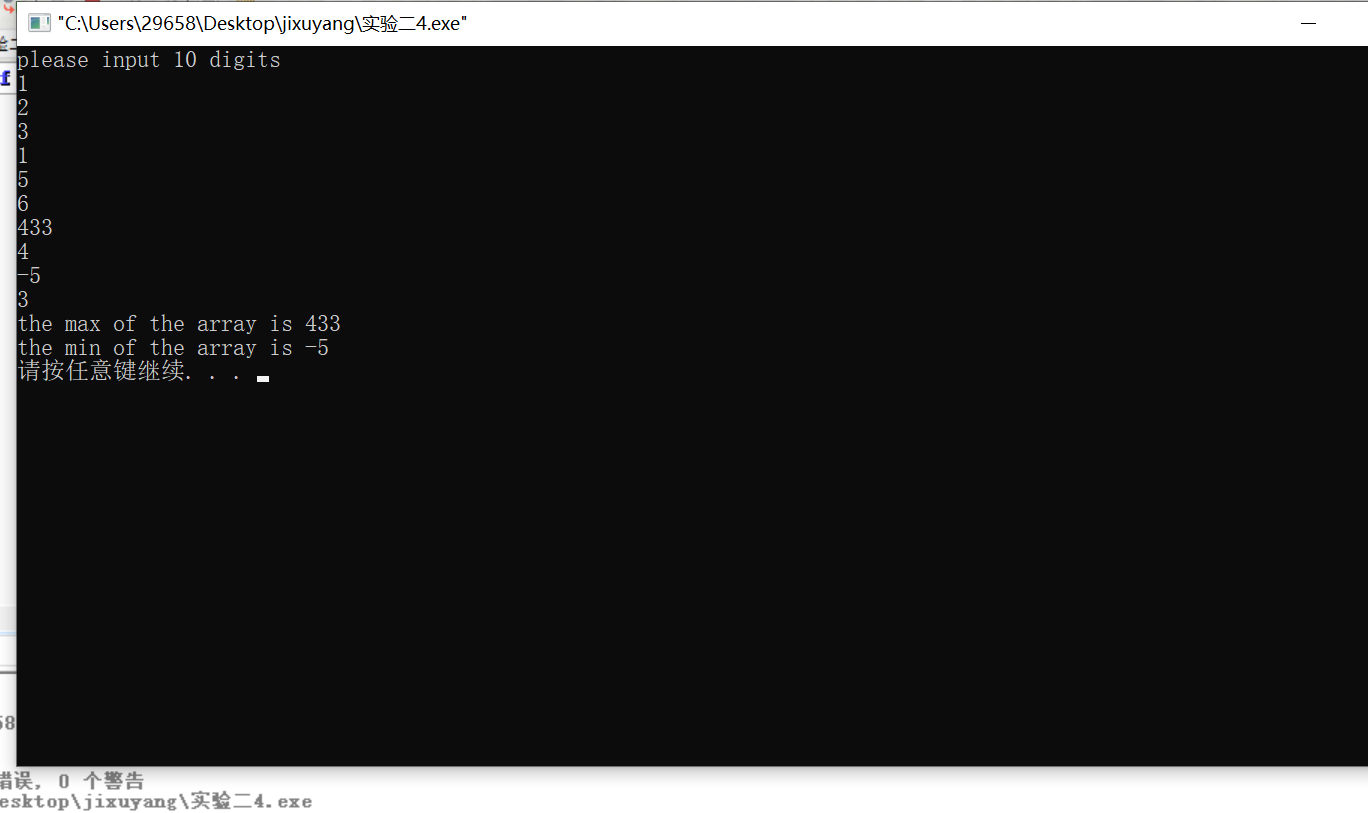
}

printf("the max of the array is %d\n",max);

printf("the min of the array is %d\n",min);

return 0;

}

验证与结论：

可以输出十个数字中的最大值与最小值。

总结与心得体会：for循环极大的减少了重复语句，极大的简化了一些重复语句，数组的运用也尤为重要，选择结构的进一步加深掌握。

实验题目二5：猴子吃桃问题。猴子第一天摘下若干个桃子，当即吃了一半，还不过瘾，又多吃了一个。第二天早上又将剩下的桃子吃掉一半，又多吃一个。以后每天早上都吃了前一天剩下的一半多一个。到第10天早上想再吃时，发现只剩下一个桃子了。求第一天共摘了多少个桃子。

设计思想：循环结构的运用，从第一天起开始计算。

实验代码及注释：#include<stdio.h>

int main()

{

int i,sum=0;

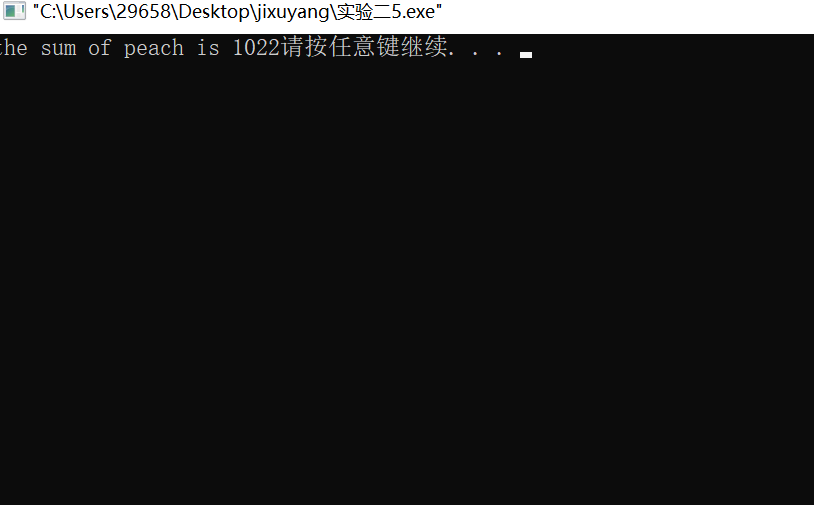
for(i=1;i<10;i++)//for循环

sum=(sum+1)\*2;

printf("the sum of peach is %d",sum);

return 0;

}

验证与结论：

直接计算出猴子第一天的摘桃数。

总结与心得体会：for循环十分重要，前提是要理清题目中的逻辑关系。

实验题目三1：输出10-100之间的所有非素数之和。

设计思想：利用数学库函数sqrt进行for循环从而判断此数是否非素数，从而进行求和。

实验代码及注释：#include<stdio.h>

#include<math.h>

int main()

{

int m=10,sum=0,i=2;

for(m=10;m<=14;m++)

{

for(i=2;i<=sqrt(m);i++)

if(m%i==0)

{

sum=m+sum;

break;

}

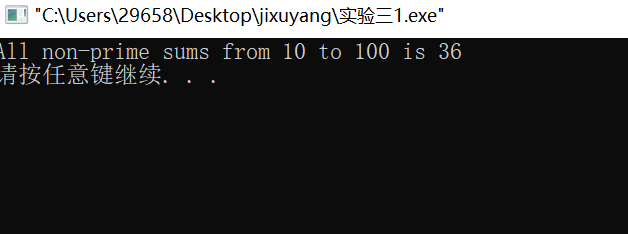
}

printf("All non-prime sums from 10 to 100 is %d\n",sum);

return 0;

}

验证与结论：



非常简单的判断了素数之和并且得出了其和。

总结与心得体会：搞清逻辑关系，学会素数的判别是第一步，再利用简单函数进行判断素数从而利用for循环进行求和得出结果，十分方便。

实验题目三2： 按顺序读入10名同学4门课程的成绩，计算出每位学生的平均分并输出。

设计思想：利用scanf语句进行读入，最后进行求和求平均数。

实验代码及注释：#include<stdio.h>

int main()

{

float score,average,sum;

int i,j;

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

{

sum=0;

for(j=1;j<=4;j++)

{scanf("%f",&score);

sum=sum+score;

}

average=0;

average=sum/4;

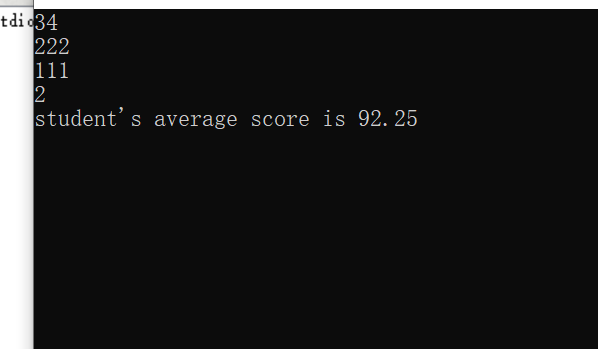
printf("student's average score is %.2f\n",average);

}

return 0;

}

验证与结论：



利用算数求和求平均数计算出最终四位同学成绩的平均数。

总结与心得体会：掌握for循环十分重要。

实验题目三3：输出以下图形：

\*\*\*\*\*H

\*\*\*\*e

\*\*\*l

\*\*l

\*o

设计思想：通过for循环与for嵌套循环交替输出“\*”和“ ”。

实验代码及注释：#include<stdio.h>

int main()

{

int i,j;

char c[5]={'H','l','l','e','o'};

for(i=0;i<5;i++)

{

for(j=5;j>i;j--)

printf("\*");

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

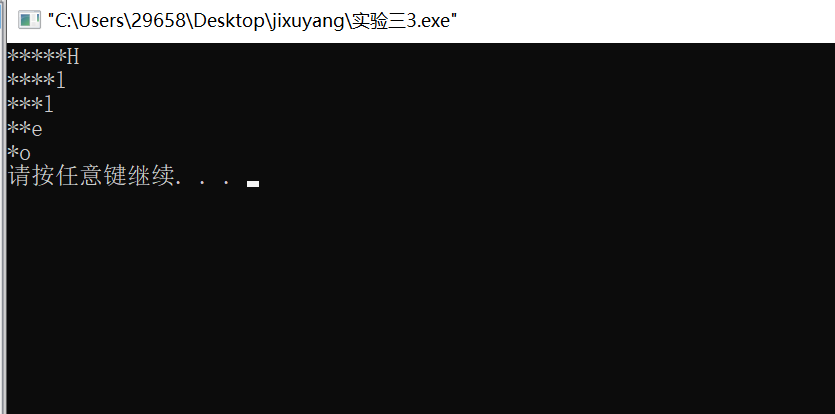
printf("");}

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

}

return 0;

}

验证与结论：

For循环中嵌套循环达到倒三角的目的。

总结与心得体会：数组与for嵌套循环还有char的字符型输出联合具有重大意义，实现多种图形与字符的输出。

实验题目三4：求1——2000之间的所有完数。完数是指一个数恰好等于它的因子之和。

设计思想：利用for循环并且嵌套最终利用选择结构if语句判断是否符合条件。

实验代码及注释：#include<stdio.h>

int main()

{

int i,j,sum;

for(i=1;i<=2000;i++)

{

sum=0;

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

if(i%j==0)

sum+=j;

if(sum==i)

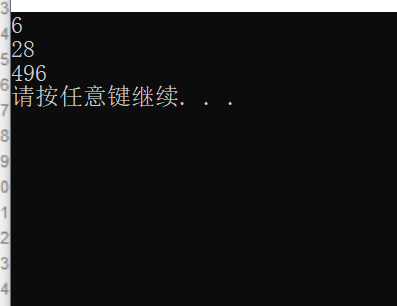
printf("%d\n",i);

}

return 0;

}

验证与结论：



输出了规定区间内的完数。

总结与心得体会：for循环的嵌套与选择结构联系可以解决各种各样的数字判断问题，需要很好的掌握。

实验题目三5：打印出所有水仙花数

设计思想：利用for循环求出规定区间内所有的水仙花数。

实验代码及注释：#include<stdio.h>

int main()

{

int s,a,b,c;

for(s=100;s<=999;s++)

{a=s/100;

b=s/10-(a\*10);

c=s%10;

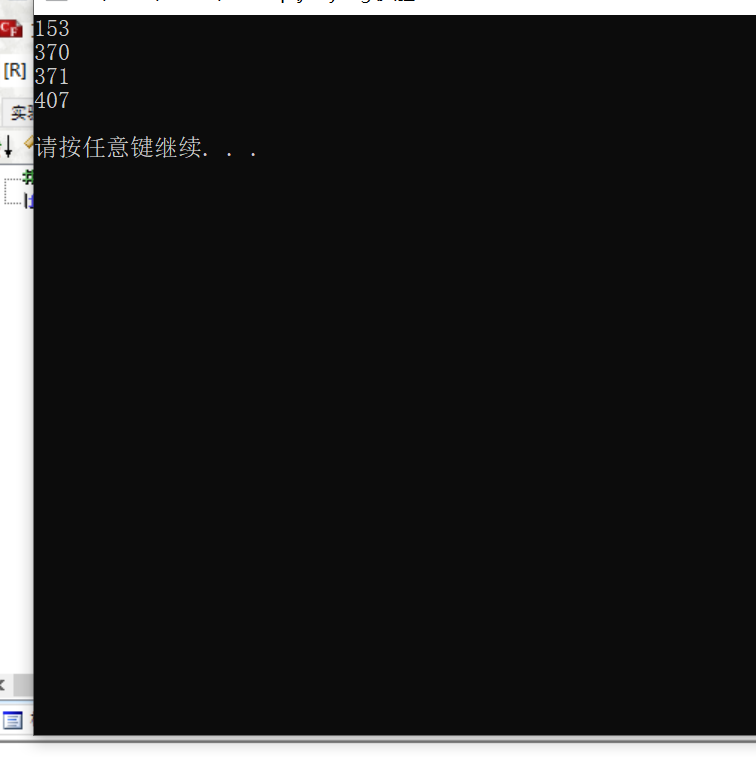
if(s==a\*a\*a+b\*b\*b+c\*c\*c)

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

printf("\n");

return 0;

}

验证与结论：

按要求输出了区间内的所有水仙花数。

总结与心得体会：for循环十分重要且要有较高的逻辑思维能力。

实验题目三6：写一个函数，求以下数列前N项之和。

2/1，3/2，5/3，8/5，13/8，21/13，···

设计思想：通过定义中间整型变量，进行fz和fm的运算求和进而输出小数。

实验代码及注释：#include<stdio.h>

int main()

{

int n,i=1;

double fz=2,fm=1,sum=0;

scanf("%d",&n);

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

{double temp=0;

sum=sum+(fz/fm);

temp=fz;

fz=fm+fz;

fm=temp;}

printf("%0.2lf\n",sum);

return 0;

}

验证与结论：

计算出前几项和，项数由用户决定。

总结与心得体会：整型变量定义，输出浮点型变量并且保留小数点后两位都是需要注意的，for循环需要熟练掌握。

实验题目四1：写一个函数，Sn=a+aa+aaa+···+aa···a

求多项式的前n项和，其中a是一个数字。n由键盘输入。

设计思想：for循环以及数学指数函数的综合运用。

实验代码及注释：#include<stdio.h>

#include<math.h>

int main()

{

int a=5,i,n,t=0,sum,m=0;

scanf("%d",&n);

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

{sum=0;

m=a\*(pow(10,i));

t=t+m;

sum=sum+t;

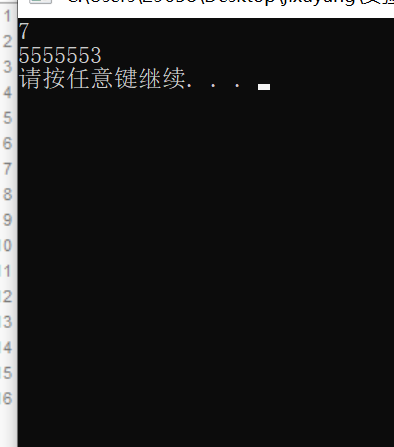
}

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

return 0;

}

验证与结论：



进行5到5^7进行求和

总结与心得体会：学会利用简单的数学函数十分重要，scanf函数必须掌握，for循环加深理解。

实验题目四2：编程实现求10000以内的回文数。其中，函数实现判断任一正整数是否为回文数。所谓回文数就是将一个数从左向右读与从右向左读是一样的，例如121和1331都是回文数。

设计思想：利用for循环判断谁是回文数。

实验代码及注释：include<stdio.h>

int main()

{

int a,b,c,m,n;

printf("please enter one number\n");

scanf("%d",&a);

if(0<a&&a<10)

printf("%d is palindromic\n",a);

else if(10<a&&a<100)

{

b=a/10;

c=a%10;

if(b==c)

printf("%d is palindromic\n",a);

else

printf("%d is not palindromic\n",a);

}

else if(100<a&&a<1000)

{

b=a/100;

c=a%10;

if(b==c)

printf("%d is palindromic\n",a);

else

printf("%d is not palindromic\n",a);

}

else if(1000<a&&a<10000)

{b=a/100;

c=a%100;

m=c/10;

n=c%10;

c=n\*10+m;

if(b==c)

printf("%d is palindromic\n",a);

else

printf("%d is not palindromic\n",a);

}

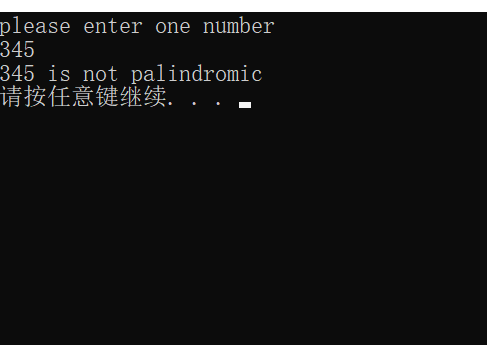
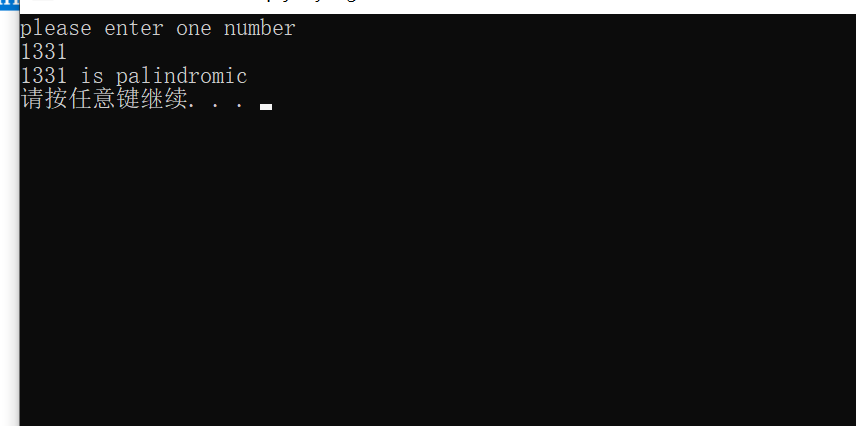
else if(a>=10000)

printf("error\n");

return 0;

}

验证与结论：

可以很正确的判断输入数字是否为回文数

总结与心得体会：if语句的妙用，但是比较复杂，也可以选择循环语句。因为本题是10000以内所以可以选择for循环。

实验题目四3：假设你每月在储蓄账户上存100元，年利率是5%，则每月的利率是0.05/12=0.00417。第一个月后，账户上的值变成100\*（1+0.00417）=100.417；第二个月后，账户上的值变成（100+100.417）\*（1+0.00417）=201.252；第三个月后，账户上的值变成（100+201.252）\*（1+0.00417）=302.507，以此类推。写一个函数，根据用户输入的每月的存款数、年利率和月份数，计算给定月份后账户上的钱数。

设计思想：定义一个中间变量利用for循环。

实验代码及注释：#include<stdio.h>

int main()

{

int month=0,i;

float yr=0.05,mr=yr/12,temp=0,money=0;

scanf("%f,%d",&money,&month);

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

{

money=(money+temp)\*(1+mr);

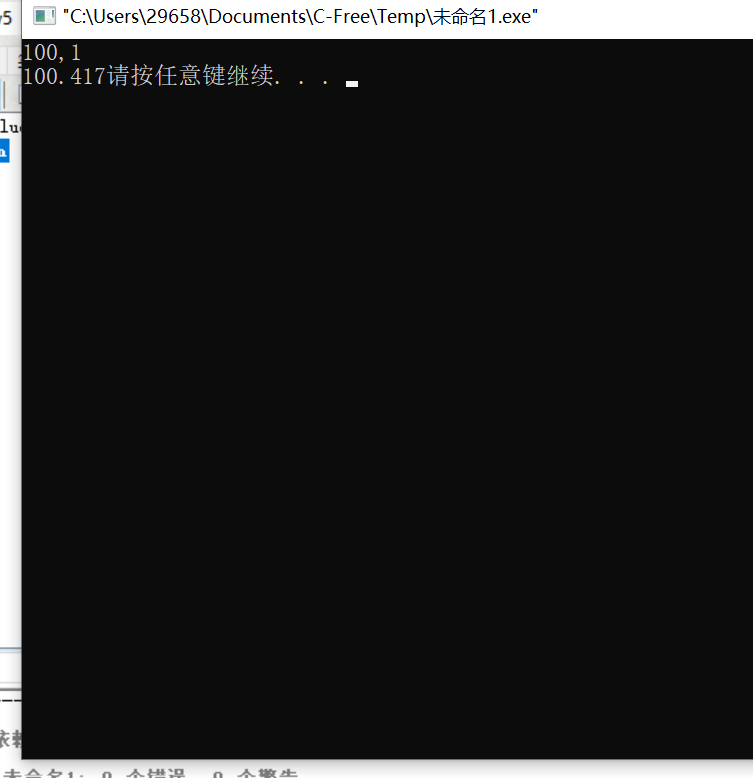
temp=money;

}

printf("%.3f",money);

return 0;

}

验证与结论：

计算出了存入的钱和所存月份后的总钱数。

心得体会与总结：学会使用中间变量并且理清逻辑关系很重要，最终结合for循环。

实验题目四4：用递归法将一个整数n转换成字符串。例如，输入483，应输出的字符串“483”，n的位数不确定，可以是任意位数的整数。

设计思想：利用自己定义的函数对数字进行处理。

实验代码及注释：#include<stdio.h>

#include<stdlib.h>

int solve(int n)

{

int temp=n%10;

int tt;

n/=10;

if(n)

solve(n);

tt='0'+temp;

if(n)

printf(" %c", (char)tt);

else

printf("%c", (char)tt);

}

int main()

{

int n;

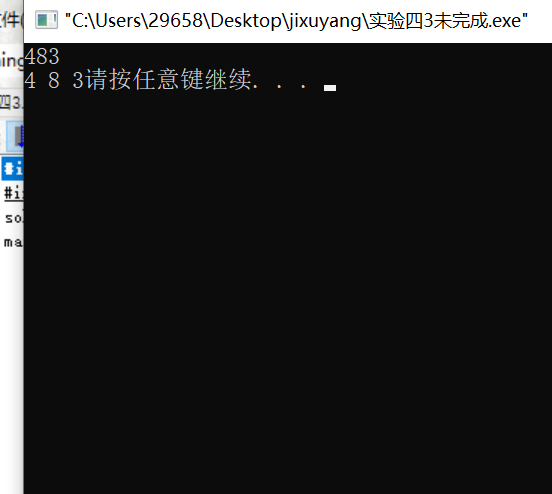
scanf("%d",&n);

solve(n);

return 0;

}

验证与结论：



可以按照题目要求输出为字符串的形式。

实验题目四5：用递归方法求n阶勒让德多项式的值

设计思想：经过自己定义的函数按照公式进行运算。

实验代码及注释：#include<stdio.h>

#include<stdlib.h>

int main()

{

int temp,n;

float n\_polynomial;

float polynomial(int,int);

printf("please enter n and temp\n");

scanf("%d,%d",&n,&temp);

n\_polynomial=polynomial(n,temp);

printf("polynomial=%6.2f\n",n\_polynomial);

return 0;

}

float polynomial(int n,int x)

{

if(n==0)

{

return (1);

}

else if(n==1)

{

return (x);

}

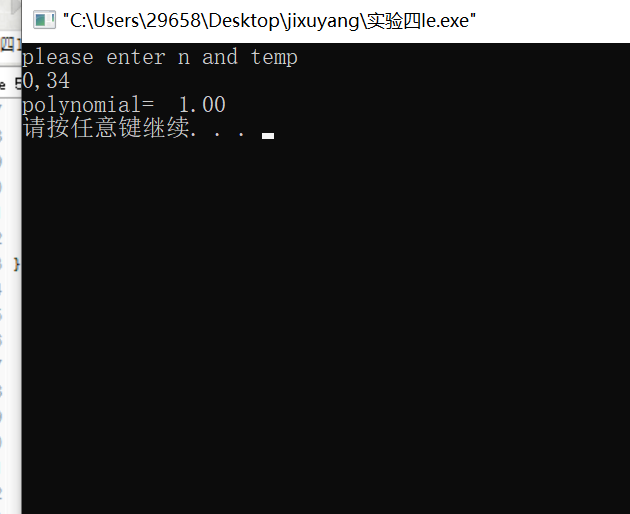
else if(n>1){

return ((2\*n-1)\*x-polynomial((n-1),x)-(n-1)\*polynomial((n-2),x))/n;

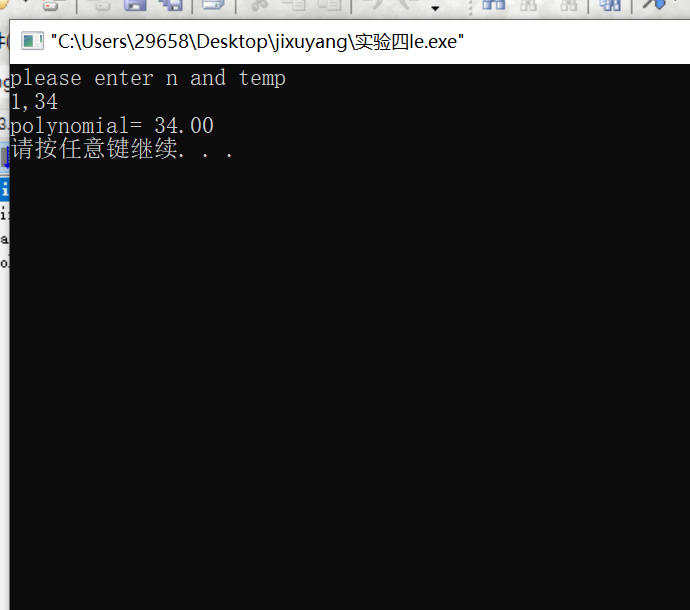
}

}

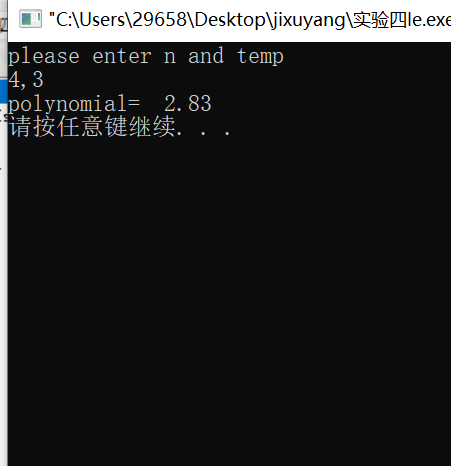
验证与结论：



n=0时，输出永远为0；



n=1时，根据公式最终结果永远是它本身。



n既不为0也不为一根据勒让德多项式计算为保留两位小数的数字。

心得体会与总结：函数不止库函数，可以根据用户需求指定符合条件的函数，函数十分重要。

实验题目四6：编程求20以内正整数的阶乘。要求：写一函数求任一整数的阶乘，在函数内部通过定义静态局部变量求阶乘。

设计思想：由于只有1的阶乘是其本身所以通过定义函数进行if进行判断最终乘到其本身。

实验代码及注释：#include<stdio.h>

#include<stdlib.h>

int fac(int n)

{

int c;

if(n==0||n==1) c=1;

else if(n>1)

c=n\*fac(n-1);

return c;

}

int main(void)

{

int i;

scanf("%d",&i);

if (i<0)

printf ("error\n");

else

printf("%d!=%d\n",i,fac(i));

return 0;

}

验证与结论：

根据scanf函数和if语句判断是否为整型正数变量，是为正确结果，不是输出error。

心得体会与总结：函数在c语言中十分重要，可以简化许多问题，但前提是搞清逻辑。

实验题目四7：编程实现两个整形变量内容互换。用带参数的宏实现变量内容的交换。

设计思想：要用宏来实现变量内容的交换，就必须要用到define

实验代码和注释：#include<stdio.h>

#define Swap(a,b){c=a;a=b;b=c;}

int main(void)

{ int a,b,c; scanf("%d%d",&a,&b);

Swap(a,b);

printf("%d%d",a,b);

return 0;

}

验证与结论：

程序运行正常，结果输出正确

总结与心得体会:交换变量不仅可以在程序主体部分实现，也可以直接利用带有参数的宏来实现，再一些大程序里面为了减少代码的数量，一般都用的带有参数的宏来实现。

**哈尔滨工程大学**

**《计算思维二》实验报告**

**基础实践二**

姓 名： 吉旭洋 班 级： 20201081

学 号： 2020108104

实验时间: 2020 年 5 月 25 日

|  |  |
| --- | --- |
| 成绩 |  |

实验题目5.1：输入两个整数数组，每个数组有五个整数，将两者和并并排列输出。

设计思想：利用scanf函数扫描之后定义数组后进行for循环进行求和运算。

实验代码及注释：

#include <stdio.h>  
#include <stdlib.h>  
int main (void)  
{  
    int a[5],b[5],c[5];  
    int i;  
    for (i=0;i<5;i++)  
    {  
        scanf("%d",&a[i]);  
    }  
    for (i=0;i<5;i++)  
    {  
        scanf("%d",&b[i]);  
    }  
    for (i=0; i<5; i++)  
    {  
        c[i]=a[i]+b[i];  
        printf("%6d",c[i]);  
    }  
    return 0;  
}

验证与结论：

可以计算出两个数组之和。

心得和体会：将数组与for循环的结合极大的简便了代码的处理。所以数组的学习也是十分重要。

实验题目五2：某公司生成5种产品，每周记录生产的每种产品数量和销售数量。每个月月末，公司将对其生产规划进行评估。该评估需要以下一个或多个信息。

每周生成和销售的数量；

所有生产产品的总量；

所有销售产品的总量；

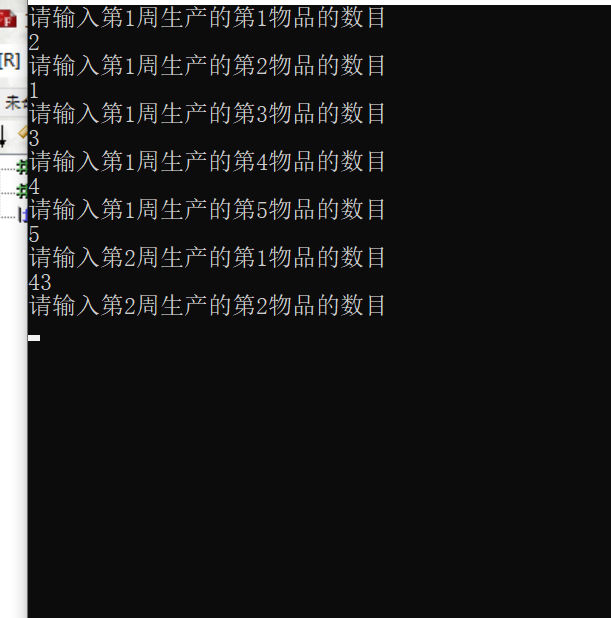
每种产品生产和销售的总量。

假设生产和销售的产品和销售分别用二维数组M和S表示，如下所示。

其中M[i][j]表示第i周生产第j种产品的数量。S[i][j]表示第i周销售第j种产品的数量。假设使用一维数组C来表示每种产品的价格。其中，C[j]表示第j种产品的价格。数组M、S和C的值都在程序中输入。

设计思想：二维数组的运用。

实验代码及注释：#include <stdio.h>  
#include <stdlib.h>  
int main (void)  
{  
    int M[4][5],S[4][5],C[5],sum1[5]={0},sum2[5]={0};  
    int i,j,sumM=0,sumS=0;  
    for(i=0;i<4;i++)  
    {  
        for(j=0;j<5;j++)  
        {  
            printf("请输入第%d周生产的第%d物品的数目\n",i+1,j+1);  
            scanf("%d",&M[i][j]);  
        }  
    }  
    for(i=0;i<4;i++)  
    {  
        for(j=0;j<5;j++)  
        {  
            printf("请输入第%d周销售的第%d物品的数目\n",i+1,j+1);  
            scanf("%d",&S[i][j]);  
        }  
    }  
    for(i=0;i<5;i++)  
    {  
        printf("请输入第%d种商品的价格\n",i+1);  
        scanf("%d",&C[i]);  
    }  
    for(i=0;i<4;i++)  
    {  
        for(j=0;j<5;j++)  
        {  
            sumM+=M[i][j];  
            sumS+=S[i][j];  
            sum1[j]+=M[i][j];  
            sum2[j]+=S[i][j];  
        }  
    }  
    printf("所有生产产品的总量为%d\n",sumM);  
    printf("所有销售产品的总量为%d\n",sumS);  
    for(i=0;i<5;i++)  
    {  
        printf("第%d种产品共生产了%d，销售了%d\n",i+1,sum1[i],sum2[i]);  
    }  
    return 0;  
}

验证与结论：

可以利用二维数组很好的完成题目给的记录问题。

心得和体会：二维数组十分重要，对实际生活中的问题具有较高的可解决性。

实验题目五3：编程将一个字符串插入到另一个字符串的第n个位置。

设计思想：数组与指针的结合，嵌入while语句实现题目要求。

实验代码及注释：#include<stdio.h>

#include<stdlib.h>

int main()

{

char a[100],b[100],\*x,\*y;//字符型数组，指针

int i;

printf("输入字符串1:");

scanf("%s",a);

printf("输入字符串2:");

scanf("%s",b);

printf("将字符串1插入到字符串2的第几个字符后:");

scanf("%d",&i);

x=a;

y=b+i;

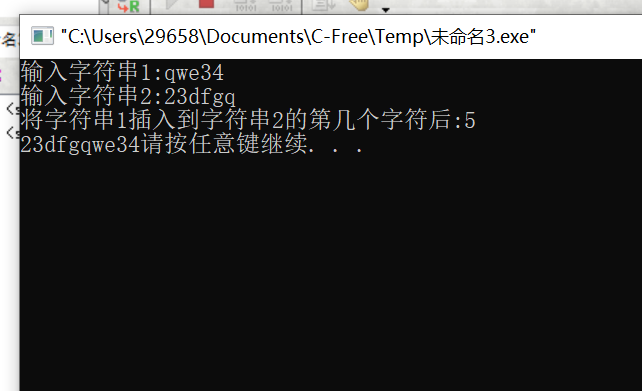
while((\*y++=\*x++)!='\0');

printf("%s",b);

return 0;

}

验证与结论：



成功将字符串一插入第二个字符串后；

心得和体会：数组与scanf函数结合十分简便，如图所示。指针的用处也十分广泛

实验题目五4：编写一个程序，使输入的一个字符串按反序存放，在主函数中输入输出字符。

设计思想：字符型数组与函数结合。

实验代码及注释：#include <stdio.h>

#include <string.h>

void reverse(char a[]);

main()

{

char str[20];

gets(str);

reverse(str);

puts(str);

return 0;

}

void reverse(char a[])

{

int i, j, k;

char t;

k=strlen(a);

for (i=0, j=k-1; i<k/2; i++, j--){

t=a[i];

a[i]=a[j];

a[j]=t;

}

}

验证与结论：

可以按要求输出字符的反串字符。

心得和体会：利用数组和字符型函数结合容易实现功能要求。

实验题目五5：编写程序，从键盘输入一行字符，统计其中英文字母、空格、数字和其他字符的个数

设计思想：定义字符变量和整型变量最后进行数组存储统计。

实验代码及注释：#include <stdio.h>

int main()

{

int i=0, space=0, num=0, n=0, ch=0;

char s[20];

printf("请输入一串字符 ");

gets(s);

while(s[i] != '\0')

{

if(s[i]==' ')

space++;

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

num++;

else if(s[i]<='z' && s[i]>='a' || s[i]<='Z' && s[i]>='A')

ch++;

else

n++;

i++;

}

printf("刚才输入的字符中英文字符个数为 %d\n", ch);

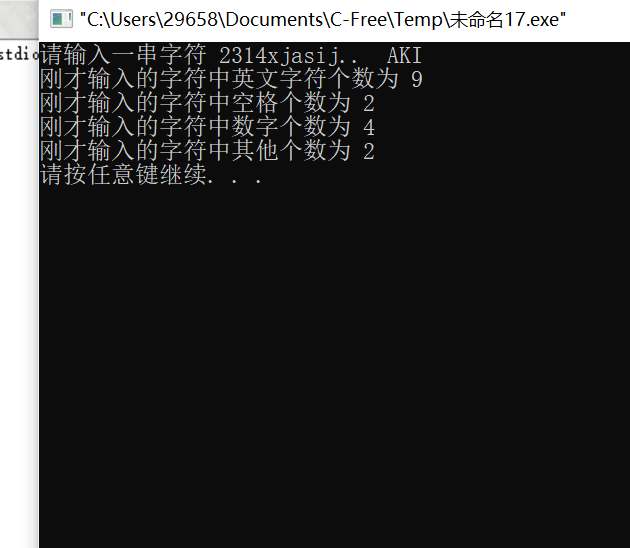
printf("刚才输入的字符中空格个数为 %d\n", space);

printf("刚才输入的字符中数字个数为 %d\n", num);

printf("刚才输入的字符中其他个数为 %d\n", n);

return 0;

}

验证与结论：

可以统计然后输出其中包含的英文字符数，空格个数，数字个数等你所需要的数据。

心得和体会：利用数组简化代码。

实验题目六1：将一个数的数码倒过来所得的心数叫做反序数，如果一个数等于他的反序数则称它为对称数，编程求不超过2000的反序数。

设计思想：利用循环语句进行1到2000的判断，选择if结构进行输出。

实验代码及注释：#include<stdio.h>

int main()

{

int a,sum=0,m,temp,s=0,p;

for(m=1;m<=2000;m++)

{

a=m;

while(a!=0)

{

sum=sum\*10+a%10;

a/=10;

}

if(sum==m)

{

s=s+1;

p=s%10;

if(p==0)

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

else

printf("%d,",m);

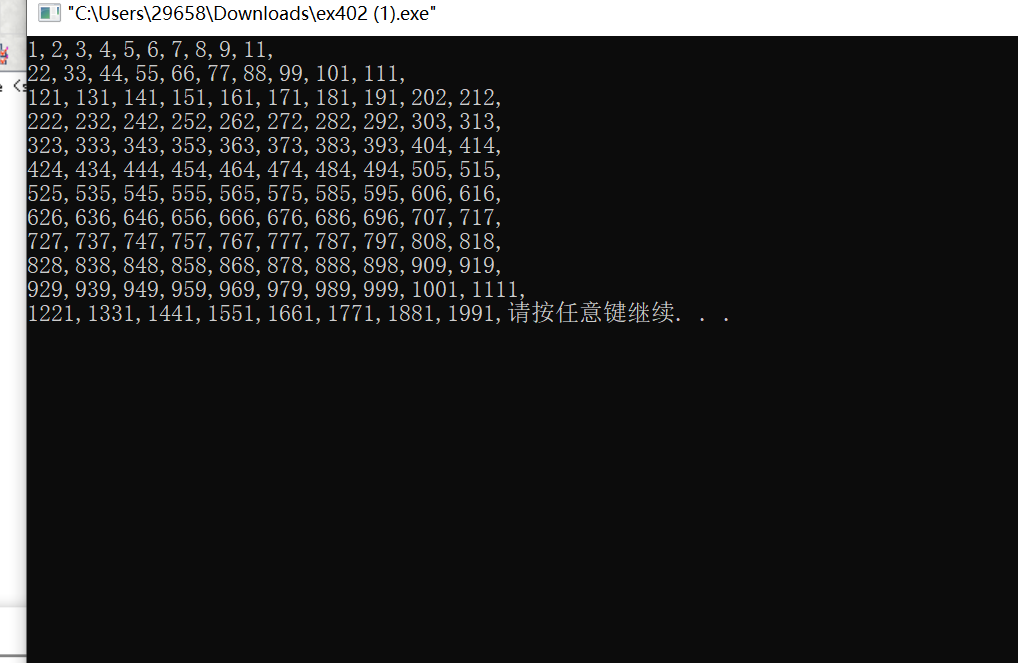
}

sum=0;

}

return 0;

}

验证与结论：

将2000以内的回文数十个十个分行输出。

心得和体会：for循环简便代码，以及保证输出的美观性和整体性，搞清各个逻辑语句之间的关系。

实验题目六2：请编写程序，判断一个数是否是回文，若是回文，则返回值为1，若不是回文，则返回值为0。

设计思想：利用while循环后利用printf函数；

实验代码及注释：#include<stdio.h>

#include<stdlib.h>

int main()

{

int n,s,y=0;

printf("please enter one number\n");

scanf("%d",&n);

s=n;

while(s>0)

{y=s%10+y\*10;

s=s/10;}

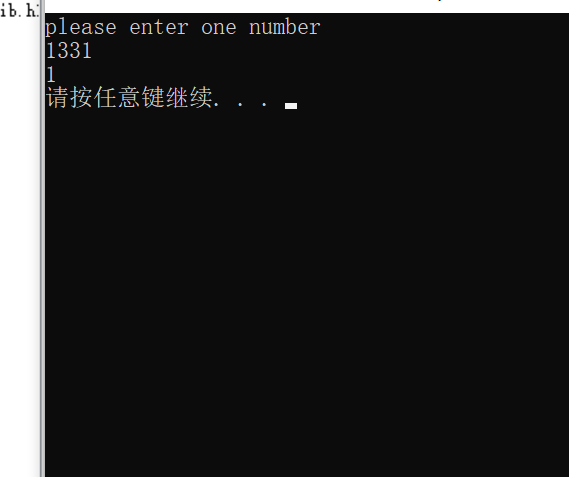
if(y==n)

{printf("1\n",n);}

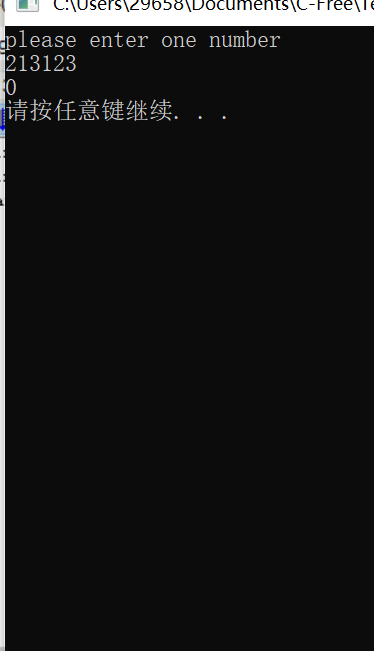
else

{printf("0\n",n);}

}

验证与结论：

是回文数返回1；



不是回文数返回0；

心得和体会：利用好循环搞清逻辑十分重要。

实验题目六3：分别用行指针和指向元素的指针求整形二维数组中的最大元素及其所在的行数和列数

设计思想：利用指针数组进行构建3行4列的数据，最后进行判断max值。

实验代码及注释：#include <stdio.h>

#define M 3

#define N 4

void Input(int (\*p)[4], int m, int n);

int FindMax(int (\*p)[4], int m, int n, int \*pRow, int \*pCol);

main()

{

int score[M][N], maxScore, row, col;

printf("Please input your data:\n");

Input(score, M, N);

maxScore = FindMax(score, M, N, &row, &col);

printf("The max is %d,in row %d,colum %d\n", maxScore, row, col);

}

void Input(int (\*p)[4], int m, int n)

{

int i, j;

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

{

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

scanf("%d", (p[i] + j));

}

}

int FindMax( int (\*p)[4], int m, int n, int \*pRow, int \*pCol )

{

int i, j, max;

max = \*(p[0]);

\*pRow = 0;

\*pCol = 0;

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

{

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

{

if ( \*(p[i] + j) > max )

{

max = \*(p[i] + j) ;

\*pRow = i;

\*pCol = j;

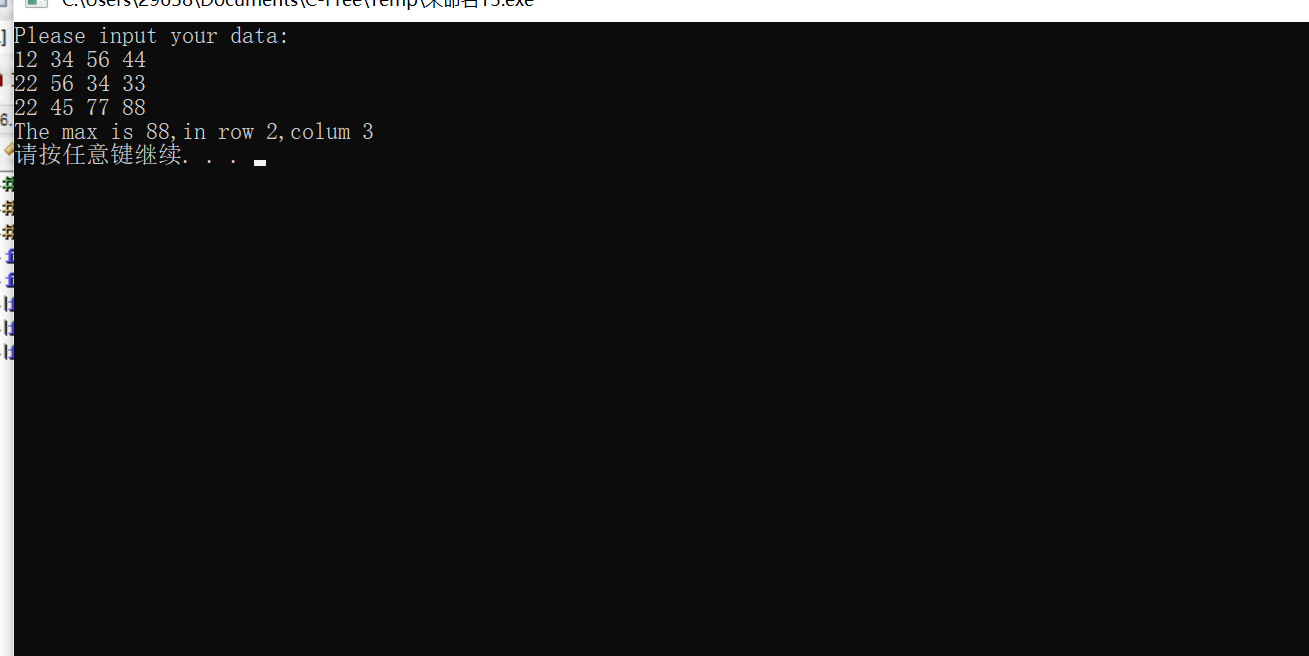
}

}

}

return max;

}

验证与结论：

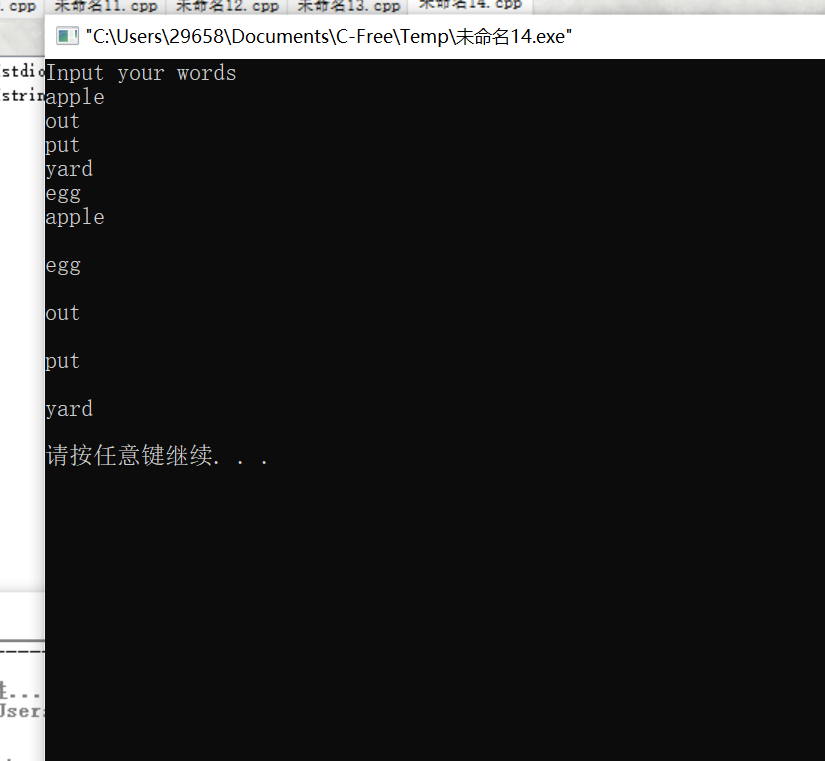
输入三行四列的12个数据最终输出最大值以及所在位置。

心得和体会：指针配合数组可以极大的简便代码，但是没有利用到结构体，可以选出我所需要的最大值。

实验题目六4：编写程序输入5个单词，将他们按照从小到大的顺序排序后输出

设计思想：利用结构体配合字符变量后进行判断最后按从小到大排序后输出。

实验代码及注释：#include <stdio.h>  
#include <string.h>  
typedef struct word  
    {  
        char Words[80];  
    }WORDS;  
void Input(WORDS \*a);  
void Sort(WORDS \*a);  
int main(void)  
{  
    WORDS data[5];  
    int i;  
    printf("Input your words\n");  
    Input(data);  
    Sort(data);  
    for(i=0;i<5;i++)  
    {  
        puts(data[i].Words);  
        printf("\n");  
    }  
    return 0;  
}  
void Input(WORDS \*a)  
{  
    int i;  
    for(i=0;i<5;i++)  
    {  
        gets(a[i].Words);  
    }  
}  
void Sort(WORDS \*a)  
{  
    int i,j;  
    WORDS temp;  
    for(i=0;i<4;i++)  
    {  
        for(j=i+1;j<5;j++)  
        {  
            if(strcmp(a[i].Words,a[j].Words)>0)  
            {  
                temp=a[i];  
                a[i]=a[j];  
                a[j]=temp;  
            }  
  
        }  
    }  
}

验证与结论：

可以按要求输出从小到大。

心得和体会：运用结构体十分重要。

实验题目六5：输入一个三位数，计算该数各位上的数字之和，如果在1到12之间，则输出与数和相对应的月份的英文名称，否则输出\*\*\*

设计思想：通过a/100将三位数的百位数字分离出来，在通过a/10-前面的乘以10后,将十位数字分离出来，再通过三位数a对10取余，将个位数字分离出来，再将各个数字相加求和用选择结构if语句。

实验代码及注释：#include<stdio.h>

int main()

{

int a,s,x,y,z;

scanf("%d",&a);

x=a/100;

y=(a/10)-(x\*10);

z=a%10;

s=x+y+z;

if(s==1)

printf("January\n");

else if(s==2)

printf("February\n");

else if(s==3)

printf("March\n");

else if(s==4)

printf("April\n");

else if(s==5)

printf("May\n");

else if(s==6)

printf("Jane\n");

else if(s==7)

printf("July\n");

else if(s==8)

printf("August\n");

else if(s==9)

printf("September\n");

else if(s==10)

printf("October\n");

else if(s==11)

printf("Novermber\n");

else if(s==12)

printf("Decmber\n");

else

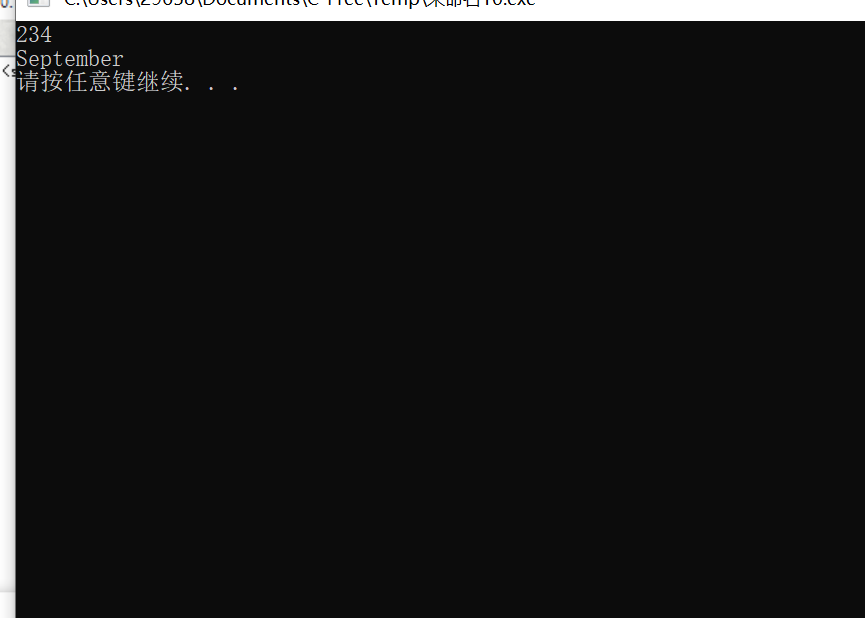
printf("\*\*\*\*\n");

return 0;

}

验证与结论：

这里是输入数据并不在1到12之间



可以得到正确输出（1<=a&&a<=12）

心得和体会：if的选择结构很重要配合逻辑语句可以很好的解决文题。

实验题目七1： 定义结构体类型，分别编写函数，实现复数的加减运算，在主函数中调用这些函数进行计算并输出计算结果。

设计思想：利用指针数组结构体进行编程达到目的。

实验代码及注释：#include <stdio.h>

#include <stdlib.h>

typedef struct complex

{

int real;

int im;

}COMPLEX;

COMPLEX ComplexPlus(COMPLEX za,COMPLEX zb);

COMPLEX ComplexMinus(COMPLEX za,COMPLEX zb);

int main(void)

{

COMPLEX x,y,result1,result2;

printf("Input x+yi:");

scanf("%d+%di",&x.real,&x.im);

printf("Input a+bi:");

scanf("%d+%di",&y.real,&y.im);

result1=ComplexPlus(x,y);

result2=ComplexMinus(x,y);

printf("(%d+%di)+(%d+%di)=(%d+%di)\n",x.real,x.im,y.real,y.im,result1.real,result1.im);

printf("(%d+%di)-(%d+%di)=(%d+%di)\n",x.real,x.im,y.real,y.im,result2.real,result2.im);

return 0;

}

COMPLEX ComplexPlus(COMPLEX za,COMPLEX zb)

{

COMPLEX z;

z.real=za.real+zb.real;

z.im=za.im+zb.im;

return z;

}

COMPLEX ComplexMinus(COMPLEX za,COMPLEX zb)

{

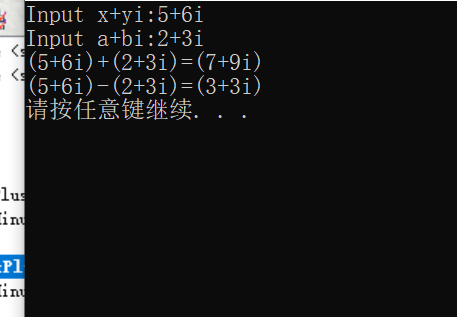
COMPLEX z;

z.real=za.real-zb.real;

z.im=za.im-zb.im;

return z;

}

验证与结论：

可以进行简单的复数运算。

心得和体会：指针具有重要功能，可以进行更的的目的要求。

实验题目七2：有10本图书，每本图书的信息包括书号、书名、作者、价格，编写函数完成以下功能：1.从键盘输入数据，将其存放在结构体数组中。2.输入书名，在数组中查找是否存在此书，有此书则输出此书的信息，无此书则输出提示信息。输入一个价格，将高于此价格的图书信息输出

设计思想：结合数组和结构体运用指针。

实验代码及注释：#include <stdio.h>

#include<string.h>

#include <stdlib.h>

typedef struct books

{

long number;

char name [80];

char writer [80];

float price;

}BOOKS;

void GetInformation(BOOKS \*a);

void printfbooks(BOOKS \*a);

int main(void)

{

BOOKS mybooks[10];

char searching[80];

int i;

float searchprice;

printf("Please input your books information\n");

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

{

GetInformation(&mybooks[i]);

}

printf("Please input the book you want to search\n");

getchar();

gets(searching);

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

{

if(strcmp(mybooks[i].name,searching)==0)

{

printfbooks(&mybooks[i]);

}

}

printf("Input the price you want to search\n");

scanf("%f",&searchprice);

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

{

if(mybooks[i].price>searchprice)

{

printfbooks(&mybooks[i]);

}

}

return 0;

}

void GetInformation(BOOKS \*a)

{

scanf("%ld",&a->number);

getchar();

gets(a->name);

gets(a->writer);

scanf("%f",&a->price);

}

void printfbooks(BOOKS \*a)

{

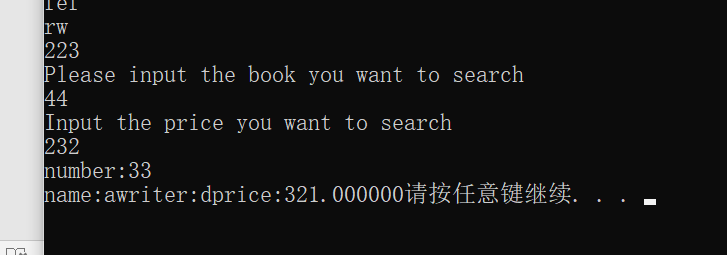
printf("number:%ld\n",a->number);

printf("name:%s",a->name);

printf("writer:%s",a->writer);

printf("price:%f",a->price);

}

验证与结论：

输入数据可以根据需求呈现出你符合要求的书。

心得和体会：指针与以前的数组和结构体结合有着巨大的用处，可以解决更多的实际问题，加紧融会贯通。

实验题目七3：设有若干个人员的数据，其中有学生和教师。学生的数据包括：姓名学号、性别、班级、，教师的数据包括：姓名、职工号、性别、职务。现要求将老师和学生的数据放在同一表格中，其中老师的职务和学生的班级共享空间。编程输入相应数据，然后再输出数据

设计思想：利用结构体指针与链表结合。

实验代码及注释：#include <stdio.h>

#include <malloc.h>

#include <stdlib.h>

#define N 4

struct studentteacher

{

char name[16];

char xgh[20];

char sex[5];

union

{

int clas;

char office[16];

}claoff;

struct studentteacher\*next;

}\*head,\*pthis;

void input()

{

struct studentteacher \*tmp;

printf("请输入个人信息:\n");

printf("-------------------------------\n");

for (int i=1;i<=N;i++)

{

printf("姓名\t学号/职工号\t性别\t班级/职务\n");

if ((tmp=(struct studentteacher \*)malloc(sizeof(struct studentteacher)))==NULL)

{

printf("\n错误!不能申请所需的内存!\n");

exit(0);

}

scanf("%s\t%s\t%s",&tmp->name,&tmp->xgh,&tmp->sex);

if(tmp->xgh[0]=='s')

{

scanf("%d",&tmp->claoff.clas);

}

else

{

scanf("%s",tmp->claoff.office);

}

tmp->next=NULL;

if (head==NULL)

{

head=tmp;

pthis=head;

}

else

{

pthis->next=tmp;

pthis=pthis->next;

}

}

}

void output()

{

printf("\n\n数据列表\n");

printf("姓名\t学号/职工号\t性别\t班级/职务\n");

printf("-------------------------------\n");

if (head==NULL) {

printf("错误,没有数据!\n");

return;

}

pthis=head;

while (pthis!=NULL)

{

if(pthis->xgh[0]=='s')

{

printf("%s\t%s\t%s\t%d\n",pthis->name,pthis->xgh,pthis->sex,pthis->claoff.clas);

}

else

{

printf("%s\t%s\t%s\t%s\n",pthis->name,pthis->xgh,pthis->sex,pthis->claoff.office);

}

pthis=pthis->next;

}

}

int main()

{

printf("\n学生学号以s开头，成员数为4人\n\n");

head=NULL;

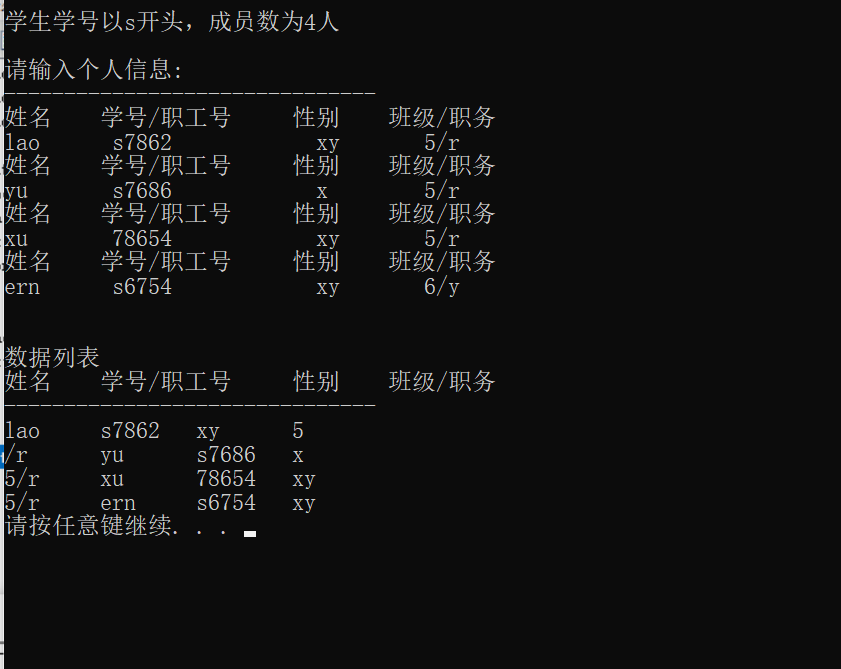
input();

output();

system("pause");

return 0;

}

验证与结论：

可以输出自己所需要的学生的各个数据。

心得和体会：指针与链表的结合极大的简便了问题，可以展示我们所需要的数据。

实验题目八1：从键盘输入一串大小写字母混合的字符串，将其中的所有小写字母转换成大写字母，大写字母转换成小写字母，并将转换后的小写字母和大写字母分别存入不同的文件中

设计思想：

实验代码及注释：

验证与结论：

心得和体会：

实验题目八2：从键盘输入4位学生的数据（学生数据包括学号、姓名和三门课成绩），然后把它们转存到磁盘文件中去，再从磁盘文件读入相应变量中，并将变量内容输出到屏幕。

设计思想：定义字符变量，整型变量利用文件写出我所需要的数据进入磁盘后再输出到屏幕，结合指针和结构体等。

实验代码及注释：#include<stdio.h>

#include<string.h>

#include <malloc.h>

#include <stdlib.h>

struct student

{

int num;

char name[32];

float score1;

float score2;

float score3;

struct student \*next;

}\*first,\*mod,s;

void input() {

struct student \*tmp;

printf("please input datas :\n");

for (int i=1;i<=4;i++) {

printf("NUM\tname\tscore1\tscore2\tscore3\n");

if ((tmp=(struct student \*)malloc(sizeof(struct student)))==NULL) {

printf("\nerror!\n");

exit(0);

}

scanf("%d\t%s\t%f\t%f\t%f",&tmp->num,tmp->name,&tmp->score1,&tmp->score2,&tmp->score3);

tmp->next=NULL;

if (first==NULL) {

first=tmp;

mod=first;

} else {

mod->next=tmp;

mod=mod->next;

}

}

}

void xieru()

{

FILE \*fp;

if((fp=fopen("student.bin","wb"))==NULL)

{

printf("can't open file\n");

return;

}

if (first==NULL) {

printf("error no datas!\n");

return ;

}

mod=first;

while (mod!=NULL)

{

if(fwrite(mod,sizeof(struct student),1,fp)==0)

{

printf("can't open file\n");

return;

}

mod=mod->next;

}

fclose(fp);

}

void duchu()

{

FILE \*fp;

if((fp=fopen("student.bin","rb"))==NULL)

{

printf("can't open file！\n");

return;

}

struct student \*tmp;

for (int i=1;i<=4;i++) {

if ((tmp=(struct student \*)malloc(sizeof(struct student)))==NULL) {

printf("\nerror!can't use neicun!\n");

exit(0);

}

if(fread(tmp,sizeof(struct student),1,fp)==0)

{

printf("can't read file！\n");

return;

}

tmp->next=NULL;

if (first==NULL) {

first=tmp;

mod=first;

} else {

mod->next=tmp;

mod=mod->next;

}

}

fclose(fp);

}

void list() {

printf("\n\data liebiao\n");

printf("-------------------------------\n");

if (first==NULL) {

printf("error,no datas!\n");

return;

}

mod=first;

while (mod!=NULL) {

printf("%d\t%s\t%.2f\t%.2f\t%.2f\n",mod->num,mod->name,mod->score1,mod->score2,mod->score3);

mod=mod->next;

}

}

int main()

{

FILE \*fp;

first=NULL;

input();

xieru();

first=NULL;

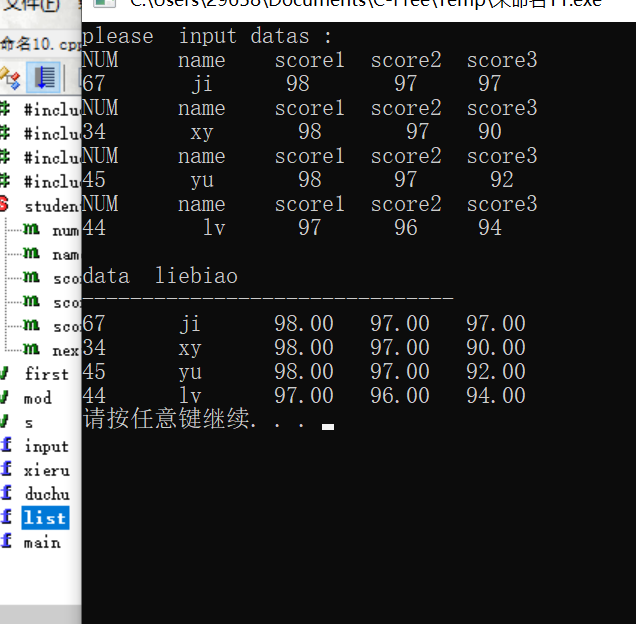
duchu();

list();

system("pause");

return 0;

}

验证与结论：

可以很好的将学生的数据存入文件再转存到磁盘中去。

心得和体会：文件的正确使用与指针，结构体的综合应用是息息相关的，我们必须很好的利用文件的综合性实现自己需要的功能。

**哈尔滨工程大学计算机基础课程教学中心**