前言

2020年10月26日

**程序整理**

黄想

20049200355

不知不觉，笔者已经进入大学半个学期。在这半个学期里，笔者学习了很多计算机编程技巧，输入与输出、判断语句、循环语句、数组、函数等等，却一直苦于没有时间对其进行整理。适逢老师要求完成一篇计算机报告，故笔者将利用此次报告对这几个月学到的计算机知识和编写的程序进行整理。在之后的报告中，将主要针对一些典型问题进行解答与思考，主要将分为题目、解答、要点、心得几个板块，其中，题目与解答将以黑色字体展示，要点用蓝色字体展示，心得用橙色字体展示。

由于时间原因及笔者水平有限，报告中难免面存在疏漏和不当之处，敬请有关专家和广大读者提出宝贵意见。

黄想

2020年10月

目录

前言 ·························································· 2

1. 输入与输出 ············································ 4
2. 判断语句 ·············································· 6
3. 循环语句 ············································· 8
4. 判断与循环结合 ····································· 11
5. 数组训练 ······································· 14
6. 输入与输出
7. 变量的赋值与输出：请输入和输出整形变量、单精度浮点数、双精度浮点数、字符、并求出其所占字节：

#include<stdio.h>

int main()

{

int a; //int表示整型变量

float f; //float表示实数

double d; //double表示较长的实数

char c; //char表示字符

scanf("%d %f %lf %c",&a,&f,&d,&c);

//%表示引用。引用之后的量，输入时&号不能少

printf("size=%d,%d,%d,%d\n",sizeof(int),sizeof(float),sizeof(double),sizeof(char)); //sizeof()就是一个求所占空间大小的小结构

printf("a=%d,f=%f,d=%lf,c=%c",a,f,d,c);

return 0;

}

输出结果：

（由于hello world过于简单，没有整理的必要，故就以此题作为全书的开端。）

心得：

1. 主要还是学会各种字符的含义和对应关系吧，int、float、double、char的含义蓝字有说明，其分别对应d、f、lf、c，除此之外，还有长整型变量long（对应ld）。
2. %表示对引号后面内容的导入或导出，%.n表示保留n位小数。
3. scanf表示输入，printf表示输出，它们都是标准输入输出格式（stdio.h）中的语言。
4. 输入计算并输出：试计算3ae/cd，其中a、e、c、d均由用户输入：

#include<stdio.h>

int main()

{

float a,e,c,d;

double y;

scanf("%f%f%f%f",&a,&e,&c,&d); //类似题1中的赋值语句

if(c\*d!=0) //利用判断语句排除cd等于0的情况

{

y=(3.00\*a\*e)/(c\*d);

printf("y = %.3lf",y); //%.3lf表示输出保留三位小数的双精度

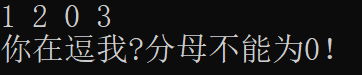
}

else printf("你在逗我?分母不能为0！")；

return 0;

}

输出结果：



心得：

1. 首先是这个判断语句，这个在之后的章节会详细展开。
2. 编程中的加减乘除可以用平时的符号计算。
3. （☆）除法中的3.00很关键，因为计算机默认整数除整数的结果为整数，故需要3.0来使得分母/分子变成小数，使得计算结果保留小数。
4. %.n表示保留n位小数，%n表示从右第n位对齐，%n.m表示保留m位小数并从右第n位对齐。
5. （取整取余）输入一个五位数，求出其各位的数字：

#include<stdio.h>

int main()

{

int shuru,a,b,c,d,e,f;

scanf("%d",&shuru);

a=shuru%10; //个位

b=shuru/10%10; //十位

c=shuru/100%10; //百位

d=shuru/1000%10; //千位

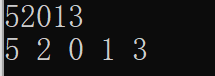
e=shuru/10000; //万位

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

return 0;

}

输出结果：



心得：

关键是明白%和/的作用，%表示取余，可以用于求末尾数字，/表示除（当为整数/整数时为整除），故可利用/10进行降位处理。在之后的学习中，还可以利用循环对该程序化简，更可以利用循环和判断的组合对任意数字进行拆分。

1. 判断语句

一、标准格式：

if-else语句： switch-case语句：

If(条件1) {执行1} switch()

else if(条件2) {执行2} {

else if(条件3) {执行3} case 值1:·····break；

······ case 值2：····break；

else{最终执行} ·····

default:······

}

两种语句对比：

1. if-else语句比较通用，可以使用各种逻辑判断和范围判断，而switch-case语句只能对某字符进行特值判断。
2. switch-case语句的处理方式是跳跃索引，而if-else语句则是顺序处理，故在处理速度上，switch-case语句比if-else语句快一些。
3. 习题心得：

1.电价分三个档次，[0,110]度电，每度电0.5元；(110,210]度电，超出110部分每度电0.55元，超过210度电，超出210部分每度电0.70元，给出一个家庭一月用电量，请计算出应缴的电费（四舍五入，保留小数点后两位小数）：

#include<stdio.h>

int main()

{

float dianliang,jiage;

scanf("%f",&dianliang);

if(dianliang>=0&&dianliang<=110) //&&表示和号，意思是同时满足时执行

{

jiage=0.5\*dianliang;

}

else if(dianliang>=110&&dianliang<=210)

{

jiage=55+(dianliang-110)\*0.55;

}

else

{

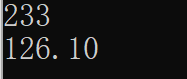
jiage=110+(dianliang-210)\*0.70;

}

printf("%.2f",jiage); //%.2表示保留两位小数

return 0;

}

输出结果：  


心得：

（i）主要还是了解了if-else结构的基本用法，并且学习了if-else语句中的逻辑语言。

（ii）if-else结构中的逻辑结构：&&表示和，两个均满足即成立；||表示与，只要满足一个就成立，并且这两符号能多个并联使用。

2.依次输入年和月，判断该月的天数：

#include<stdio.h>

int main()

{

int nian,yue,ri;

scanf("%d%d",&nian,&yue);

if(yue==1||yue==3||yue==5||yue==7||yue==8||yue==10||yue==12)

//利用||号进行“并联”

printf("31");

else if(yue==4||yue==6||yue==9||yue==11)

printf("30");

else

{

if((nian%4)==0&&(nian%100)!=0)

printf("29");

else if((nian%400)==0) //这里的排除法很巧妙！！！

printf("29");

else

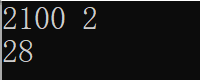
printf("28");

}

return 0;

}

输出结果：



心得：

1. 学会利用||和&&进行并联逻辑判断。
2. 学会利用集合进行巧妙设置判定选项。
3. 循环语句
4. 标准格式：  
   while语句： for语句：

while(运行条件) //满足运行条件 for（起始条件；运行条件；循环过程）

{ 时继续循环。 {

执行句 执行句

} } //起始条件是定义

比较：他两好像没啥区别？？

1. 习题心得：

1.求出斐波那契额数列：  
#include<stdio.h>

int main()

{

double fx,fxx,fxxx;

int i,j;

printf("f1=1\nf2=1\n");

fx=1;

fxx=1;

for(i=1;i<=100;i++) //for循环，先定义i=1，在i<=100时运行，每运行一次i++

{

fxxx=fx+fxx;

fx=fxx;

fxx=fxxx; //对数列进行逻辑嵌套

j=i+2;

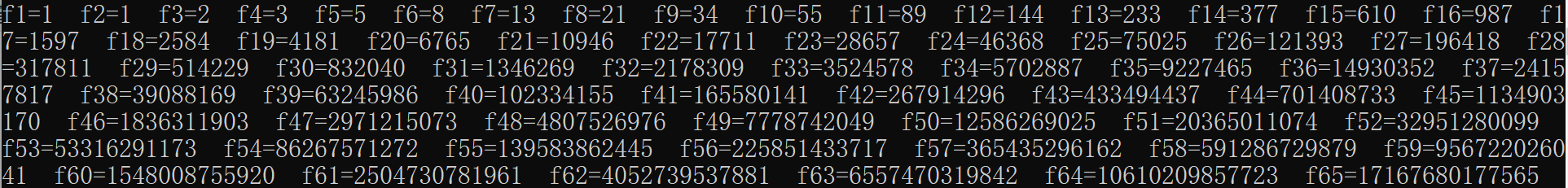
printf("f%d=%.0lf\n",j,fxxx);

}

return 0;

}

输出结果：



心得：

1. 学会使用for语句，另外，在for后的括号里，可以有多个条件，他们用逗号进行分隔，也可以没有条件，但若运行条件为空，后面应用判断语句+break进行终止循环。
2. 循环里完全可以用任何能想到的东西，就比如上面斐波那契数列的项之间的加减乘除转化，就是可以设置一个参数进行辅助转化。

2.算分小程序：老师输入人数和每个人的分数，计算其班上的平均分：

#include<stdio.h>

int main()

{

int sum,a,i,renshu;

float pingjun;

printf("算平均分小程序\n请输入考试人数\n");

scanf("%d",&renshu);

sum=0;

i=1;

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

{

printf("请输入第%d位同学的成绩\n",i);

scanf("%d",&a); //利用循环进行不断输入数据并立刻使用

sum=sum+a; //在使用a后再在之后的循环中刷新

}

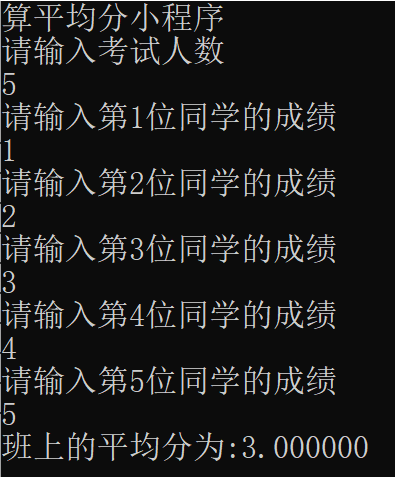
pingjun=sum\*(1.0/renshu);

printf("班上的平均分为:%f",pingjun);

return 0;

}

输入结果：



心得：

1. 利用循环进行重复输入和刷新是最骚的，比较考验对刷新变量的理解吧。
2. 不过这个用到极致的我觉得还是之后的跳一跳程序，那个是真骚！！！
3. 求出100与200之间的所有素数：

#include<stdio.h>

int main()

{

int i1,i2,a;

for(i1=100;i1<=200;i1++) //第一层循环：对100到200的数进行依次提取

{

for(i2=2;i2<=(i1-1);i2++) //第二层循环：对其进行素数判断，判断条件为其取余大于1小于它的所有数的结果都不为0

{

a=i1%i2;

if(a==0)

{

printf("%d不是素数\n",i1);

break; //只要判断出一个可取的余数就终止循环，防止多次输出和资源浪费

}

}

if(a!=0)

{

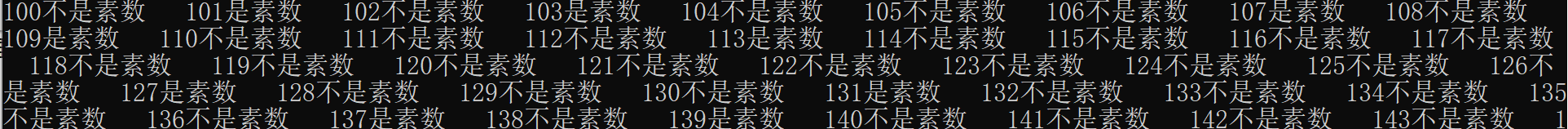
printf("%d是素数\n",i1);

}

}

return 0;

}

输出结果：  


心得:  
（i）关键是理解循环套循环的结构和对素数的判断原理。

（ii）了解break和continue，并对其进行区分：当运行到break时，整个循环终止，而当运行到continue时，该层循环终止，即次循环中continue后面的内容将不运行，进入下一层循环。

1. 循环与判断结合
2. 已知除1和其本身的所有因数之和为其的数为完数，现任意给定一整数范围，找出范围内的所有完数：

#include<stdio.h>

int main()

{

int a,b,c,i,d,sum;

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

for(i=a;i<=b;i++) //根据输入范围构造循环

{

sum=0;

for(c=1;c<=i-1;c++)

{

d=i%c;

if(d==0) //筛选因子的判断

{

sum=sum+c; //利用sum储存所需值并刷新i

}

}

if(sum==i) //在循环后判断是否为完数

{

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

}

}

return 0;

}

运行结果：  


心得：

1. 其本身就是个双重循环，并在第二层循环中利用判断进行使用储存，可以说也是一个很好的学习循环与判断结合的例题。
2. （☆）程序中红色的sum=0非常重要！！！一定要注意它的位置，不能将它放在循环外，而应该将其放到循环内，起到“初始化”的作用，即每次循环后将sum归零，防止上次循环数据留在sum中干扰之后的判定。
3. 任意输入两个数，求其最大公因数：

#include<stdio.h>

int main()

{

int a,b,c,d,e,i;

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

for(i=a;i>=1;i--) //从大到小找a的所有因数

{

c=a%i;

if(c==0)

{

if((b%i)==0) //将找到的因数代入b中看其是否为b的因数

{

printf("%d",i);

break;

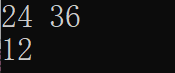
}

}

}

return 0;

}

运行结果：  


心得：

1. 这道题很容易有另一种思路，就是分别求出每个数的所有因数，再进行筛选选出其中的最大公因数。记得当时写这道题的时候觉得这种思路非常难，现在学了数组，感觉这种思路非要完成也不难。可以建立两个数组，每个数组存储其中一个数的全部因数，然后再利用双层循环，拉出一个数组中的数，与另一个数组找相同，从大到小找，也能求出最大公因数。
2. 而上面的思路则是在思考（i）时突发奇想想到的，先从大到小找a的因数，再判断找到的因数是否为b的因数，而其得到的第一个数即是其最大公因数。这个思路也给了我们做“平行类”题的另一种思路，就是将其看作非平行关系，然后选择一条线进行判断，判断成功后代入另一条线，得到结果。
3. 求公约数还有一个更简单的办法，让更小数对更大数取余，若取余为0则输出，若不为0，则让余数取余上一层的最小数，如此循环，知道取余为0，此时的取余数就是最大公因数。

3.跳一跳：即微信程序跳一跳的算分程序，使用者以空格分开的方式输入一串数字，这些数字只能是0、1、2三种情况，其中出现0则游戏结束，若为1则总分加1，若出现2则总分加二，连续出现2每次多加2（及+2+4+6+8···），是算出其总分：  
#include<stdio.h>

int main()

{

long i,a,sum,beishu;

sum=0;beishu=2;

for(i=1;i<=1000;i++)

{

scanf("%d",&a); //利用循环不断输入a

if(a==0) break;

else if(a==1) //对a进行判定加分

{

sum=sum+1;

beishu=2; //这个倍数实则加数积累，这个设置很巧妙！

}

else

{

sum=sum+beishu;

beishu=beishu+2;

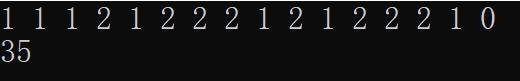
}

}

printf("%ld",sum);

return 0;

}

运行结果：  


心得：

1. 这种循环形式的输入其实有两种输入形式，一种是每输一个回车一次，还有一种是写一串，并将其用空格分开，等到输完后回车，将其同时依次输入。
2. 程序中的beishu设置地很巧妙，在连续2的执行过程中累加，而在1时将其还原成2，即相当于一次初始化过程，牛逼！

第五章 数组训练

【一叶知秋】

起因：第三次上机测试上，本来是考循环的，但是看了几眼书的我总觉得数组写这道题快然后从此踏上不归路......不过吧，以下做法的全部技巧都是自己现场想出来的，也可以说有很多不错的方法吧~

题：小明想选一个新的qq号，选号法则是：

1. 选择所有号码中各位数字之和最大的号码。
2. 若有多个号码各位数字之和相同，则选择数值最大的号码。

之后小明会输入号码的总数和各个号码。

#include<stdio.h>

int main()

{

int n,i,j,k,max1,max2;

k=0;max1=0;max2=0;

scanf("%d",&n); //输入n确定数组位数

long a[n],b[n],c[n]; //定义三个长整型数组（定义中的括号内必须是确定值）

for(i=0;i<=(n-1);i++) //实现初始数组a[n]的输入

{

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

}

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

{

c[i]=a[i]; //复制数组a[n]用于之后运算

b[i]=0;

for(j=1;j<=9;j++) //求出数组各位数字之和b[n]

{

b[i]=b[i]+c[i]%10;

c[i]=c[i]/10;

}

}

for(i=0;i<=(n-1);i++) //利用b[n]和冒泡原理找到最大值

{

if(b[i]>max1) max1=b[i];

}

for(i=0;i<=(n-1);i++) //利用循环和判断求出有几个最大值用于之后定义

{

if(b[i]==max1)

{

k++;

}

}

long d[k]; //定义新的数组用于之后的提取

j=0;

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

//这个是最骚的，将一个数组的满足条件的项导入另一个数组

{

if(b[i]==max1)

{

d[j]=a[i];

j++; //利用数组做出的双循环结构

}

}

if(k==1)

printf("%ld",d[1]);

else

{

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

{

if(d[j]>max2) max2=d[j]; //冒泡原理提取最大值

}

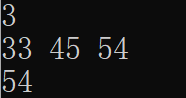
printf("%d",max2);

}

return 0;

}

运行结果：



心得：额.....这些小技巧就看上面蓝字吧.....对了！这道题的循环非常不错！！时叶彤牛逼！！！