9-21~10.21 计算机语言学习报告

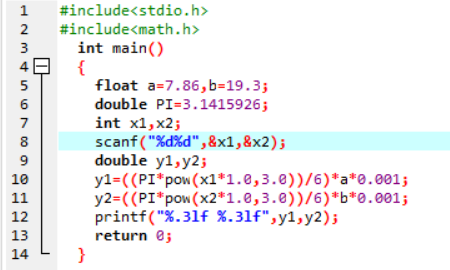
机器人工程卓越班 陈品庄 20049200264

学习计算机语言已经一个月时间了，上了七八次课，虽然每次课只有短短的一个半小时，但是学习到了很多新的知识，对编程有了最基本的了解。由于没什么基础，学起来还是感到有些不顺畅，对于一些细节理解的并不到位，虽然可能暂时能解出那些简单的题目，但是或许以后就会感到棘手了。所以一个简短的总结是必要的，既可以帮我更好地巩固一个月来的所得，又可以为以后进一步的学习打下更扎实的基础。

由于在第一次上机时提前了一个小时离开，所以不得不提前进行报告的撰写，我在这先自我检讨一下，并表示抱歉。但是当时提前离开机房，是因为看到前面走了好多同学，才走的。老师，你只针对早退的我们提出了要求，却对那些在九点甚至10点才进机房的同学视而不见，我是有点无法接受的。迟到某种程度上比早退要更恶劣吧。但我还是表示由衷的抱歉和后悔，以后上机我会充分利用时间的。

下面进入正题，我会根据近期写的题目，总结出一些小点，让以后的自己好好注意，更好地掌握这门课程。

1. 计算小球的质量

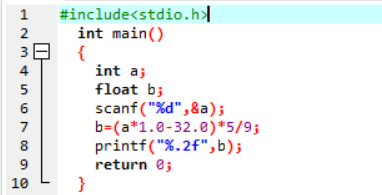


Note1.在定义像π这样的常量时 可以在程序的最开始使用define

Note2.在使用了pow这样的函数时，在程序开头得有#include<math.h>

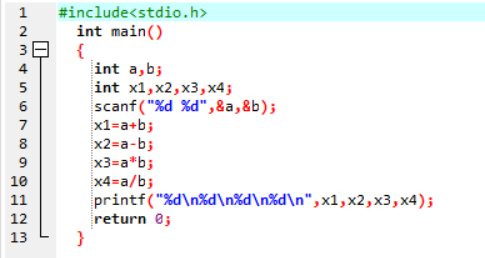
Note3.比如计算体积时，运用公式的时候不要进行化简，这可能会出现精度的问题，同时也失去了编写程序的意义

Note4.pow函数内的两个是都得是double

1. 简单的计算

Note1.我在第一次写的时候写成了（a-32）\*(5/9)，这里有蛮多细节要注意的，一个是5/9算出来并不是我所需要的，所以不能用括号括起来。第二个问题是定义的a是整型而b是单精度浮点数，所以直接写a-32是不对的，要调整。

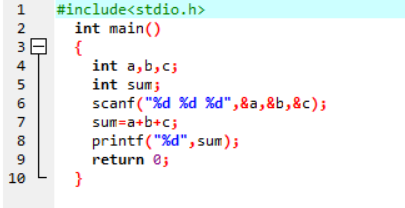
3.简单的计算



Note1.定义的变量越少越好，比如这道题中x1,x2,x3,x4完全不需要，可以直接在printf中进行设计，达成题目的要求

Note2.使用\n时两个代表字符之间不需要任何其他的东西，像空格啊，逗号，这些都不用

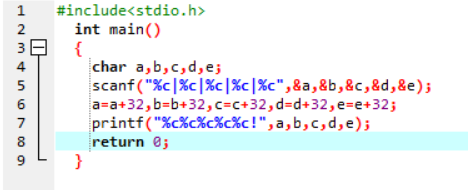
4.简单的计算



Note1.一个很平常的问题，scanf中要加&

Note2.一个很好的习惯。平常在定义变量时，可以使用number,sum，amount这些代表一定含义的词来定义，以防因为变量太多，弄出清楚每个变量的含义。

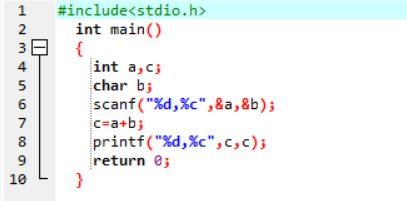
5.字符的转化



Note1.scanf(“原样输入”，......)

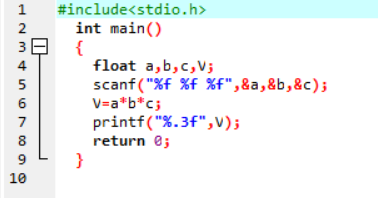
Note2.小写字母的ASCⅡ编码为大写字母的ASCⅡ编码+32。而且直接加减即可。

6.数字字符的运算



Note1.(整型数值与字符混合运算)字符对应的ASCⅡ编码参与整数运算，其结果也是整数，注意‘0’与0不同，变量a的值就是‘0’对应的ASCⅡ编码，即整数48，b就是整数0

7.实数的运算



Note1.（精度问题）32编译器：a\*b\*c与a\*c\*b结果一致，但是在64位编译器中，二者不一致。因此，浮点数运算会存在精度问题，不要随意改变运算顺序。

以上题目是第一次上机的题目，我以一个刚刚入门的萌新角度，提出一些容易被我自己忽略的问题，以及不太好理解的细节，希望能对自己往后的学习有所帮助。以上的点都是针对各个题目而言的，接下来点出一些一直让我头疼的普适性问题。

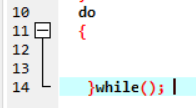
Note1.int main()后面并没有分号

Note2最后记得打return 0.

Note3.return 0 后面得有分号

Note4.{}打完花括号，换个行，让程序看起来舒心

Note5.While（）后面没有分号的 for啊之类的后面都没有，但是有一个例外的要记住

Do while中的while后面一定要加； 可以记住。

Note6.关于整数之间的除法这也是经常碰到的问题，像5/9之类的不要用括号括起来，不然会出大问题。

完蛋了，词穷了。。。这可咋整，再看几个初期的问题好了。。。

**一．**

#include<stdio.h>

int main()

{

int i=-1,a=2;

float sum=1.0;

while(a<=100)

{

i=-i;

sum=sum+1.0\*i/a;

a++;

}

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

return 0;

}

这道题我一开始是纸上谈兵的，后来上课讲的时候发现自己错误百出，首先就是把sum定义成了int。其次还写了这样一个式子（1/i）\*（-1）∧（i+1）这样一个数学表达式，但这在C语言中好像是不合法的。再有一个问题就是怎么实现这样一个循环，i=-i是一个非常巧妙的方式，而不要只记得数学中常用的（-1）的n次方来实现。这道题虽然不是很难，但是有很多细节需要注意，比如sum=sum+1.0\*i/a这个式子要注意数据类型的统一，不然无法得出想要的结果。

二．输出 100∼200 之间的不能被 3 整除的数

#include <stdio.h>

int main()

{

int n;

for (n = 100;n <= 200;n++)

{

if (n%3==0) continue;

printf(”%d␣”,n);

}

printf(”\n”);

return 0;

}

在老师上课讲到这道题的时候 我就有一种眼前一亮的感觉，如何把不能被三整除的数给循环着表达出来，这道题给了一个很好的样板。continue意味着终止本轮循环，开始下轮循环。这里需要注意的是，continue虽然终止了本轮循环，但是for语句中的i++依然是会执行的。可以死记住。通过一个if和continue就可以很好地剔除掉3的倍数。这道题的想法是挺妙的。

1. 这是一个有趣的古典数学问题: 有 一对兔子，从出生后第 3 个月起每个月都生一对兔子。小兔子长到第 3 个月后每个月又生一对兔子。假设所有兔子都不死，问每个月的兔子总数为多少？



#include <stdio.h>

int main()

{

int f1=1,f2=1,f3;

int i;

printf(”%12d\n%12d\n”,f1,f2);

for(i=1; i<=38; i++)

{

f3=f1+f2;

printf(”%12d\n”,f3);

f1=f2;

f2=f3;

}

return 0;

}

这道题里的递推是值得好好学习。f1=f2;f2=f3, 就把进程往前推了一部，可以借鉴。

小结：学习报告写到这依然感觉到几分意犹未尽。但是毕竟经历有限，一个月的学习，让我对C语言这门课程有了初步的理解，也能开始写一些简单的程序。我有几个在西交和浙大的同学，竟然还在上大计基这样愚蠢的课程，可笑可笑。有些基础知识是没有意义的，亲身实践去多写多看多练，才是最好的提升编程实力的方式，希望自己在今后的学习中，能更进一步，天天向上。