分析上机练习注意点

主 函 数 结 束

西安电子科技大学

2020

课程学习报告

胡辉帆

20049200357

机电工程学院 20200306

7

目录

1. 前言------------------P3
2. 如何学习C语言--------P3
3. 知识点-----------------P4-P15
4. 分析上机练习注意点----P16-P19
5. 后记------------------P20

课程学习报告

前言：

随着社会的不断发展，计算机的重要性越来越凸显，伴随着C语言在全世界普及推广，掌握程序设计技能也越来越重要。而进行程序设计，必须用一种计算机语言，C语言是计算机语言中最基本最关键的，故而学习C语言非常必要。

通过这段时间的学习，我对这门课有了一定的了解，以下是我是一些心得体会。

一．如何学习C语言

1.着眼于能力培养

2.掌握基本要求，注意打好基础

3.十分注重实践环节

4.举一反三，触类旁通

5.提倡和培养创新精神

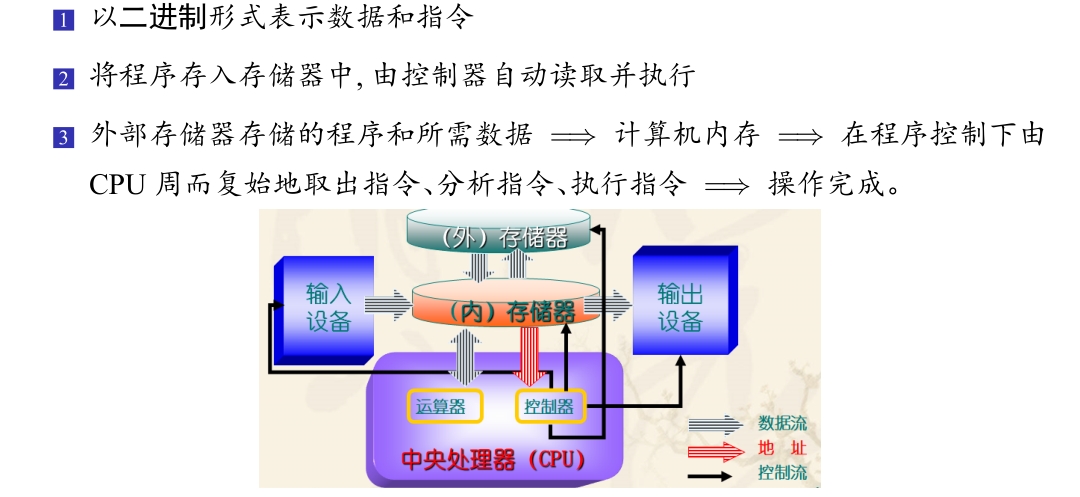
二．基础知识点

1.计算机工作原理

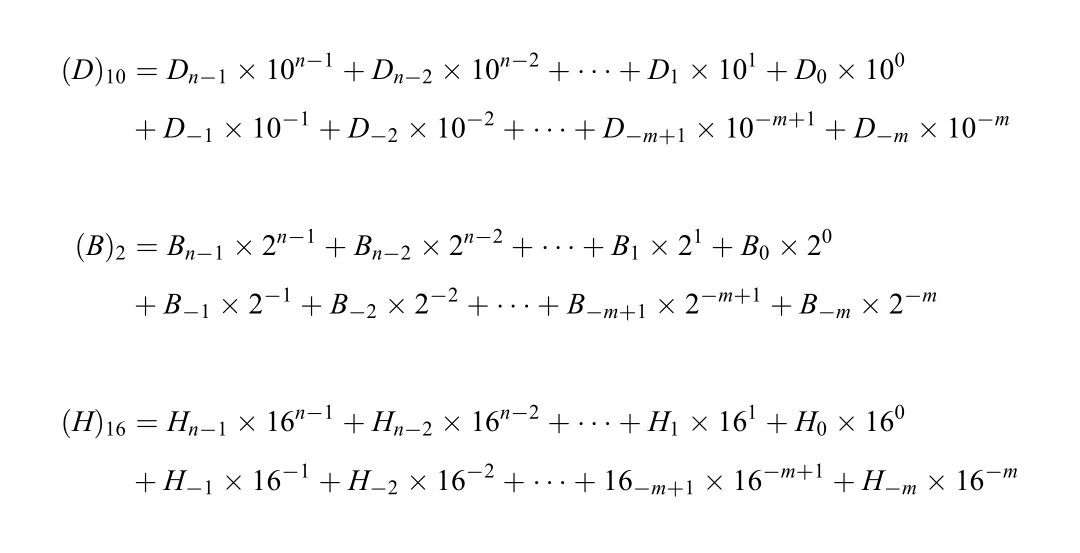
1）以二进制形式表示数据和指令

2）将程序存入存储器中，由控制器自动读取并执行

3）外部存储器存储的指令和所需数据——计算机内存——在程序控制下由CPU周而复始地取出指令、分析指令、执行指令——操作



2.十进制，二进制与十六进制

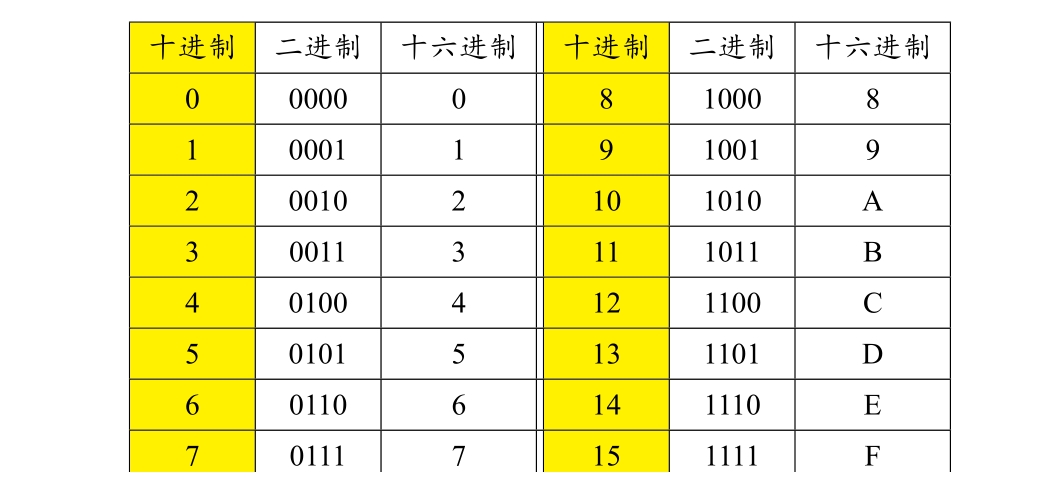


（77）10=（0100 1101）2

（77）10=（4D）16

（123）10=（7B）16

进制对照表



进制对照表对学习C语言有很大的帮助。

3.数值在计算机中的表示

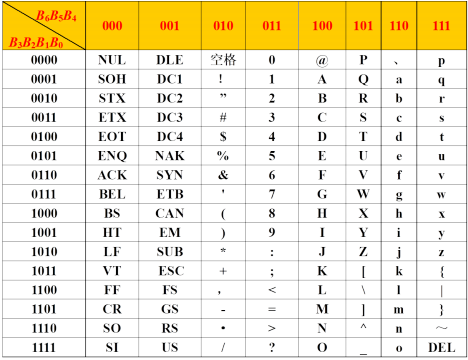
原码：，正数的符号为0，负数的符号为1，其他位按一般的方法表示数的绝对值。

反码:正数的反码与原码相同；负数的反码是符号位不变，其他位取反。

补码：正数的补。码与原来相同；负数的补码为其反码加一。



4.ASCII编码表



其中：

数字=编码值-‘0’。 9=‘9’-‘0’

大小字符间隔 ‘小’-‘大’=32

例如：

‘a’=01100001=61H=0X61=97

‘A’=01000001=41H=0X41=65

三．C语言程序设计

A.课堂上学习的第一个C语言程序

#include<stdio.h>//编译预处理指令

int main()//主函数

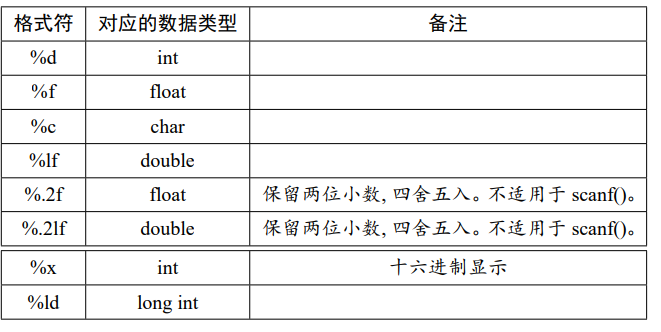
{//函数开始标志

printf(“Hello World!”);//输出一行信息

return 0;//函数执行完毕返回0

}//函数结束标志

常用格式描述符与数据类型的对应关系



ps ： 变量在使用之前首先要定义它的数据类型

int—整数 float—单精度浮点数

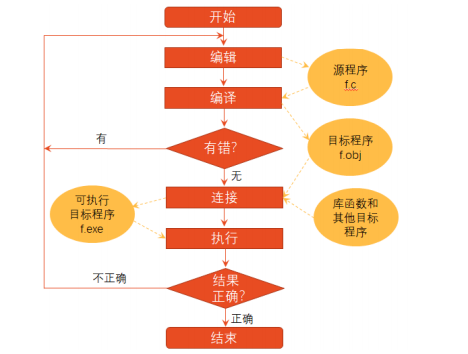
double—双精度浮点数

char—字母 sizeof—字节为

printf是输出语句 scanf是输入语句

&为取地址符，scanf语句必然存在

B.运行C程序步骤和方法



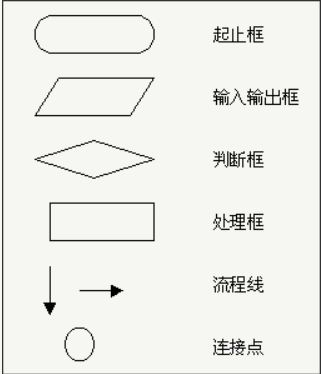
C．算法——程序的灵魂

1.算法：广义上讲，为解决某个问题而采取的方法和步骤。例如计算几个数的加减，一个方程的根，一个函数的定积分……

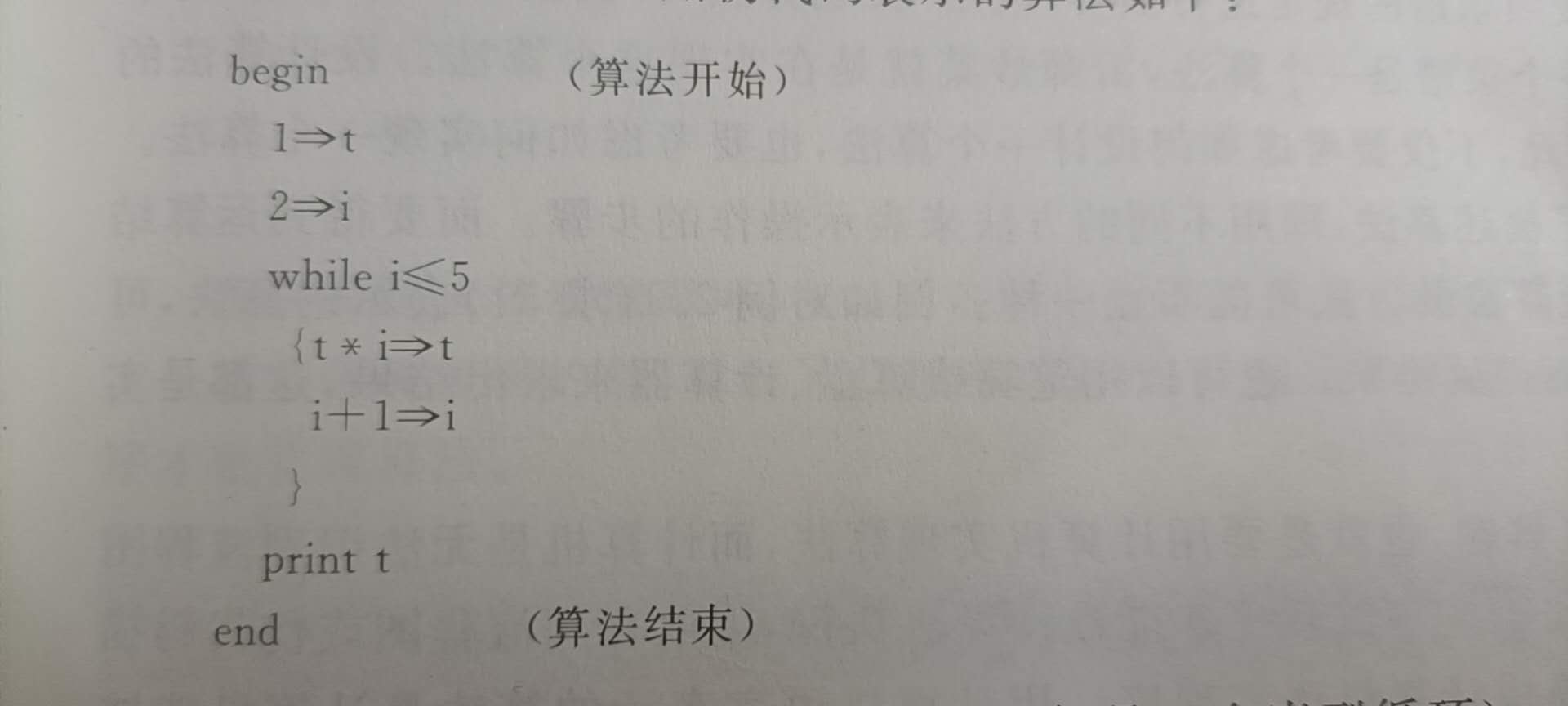
2.程序=算法+数据结构

3.算法的特性：有穷性；确定性；有零个或多个输入；有一个或多个输出；有效性。

4.用流程图表示算法



5.用伪代码表示算法（求5的阶乘）



6.用计算机语言表示算法（求5的阶乘）

#include<stdio.h>

int main()

{

int i,t;

t=1;

i=2;

while(i<=5)

{

t=t\*i;

i=i+1;

}

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

return 0;

}

D.选择结构与循环结构设计

1.if(条件表达式)

#include<stdio.h>//预处理指令

int main() //主函数

{ // 函数开始标志

int a=10; // 定义变量a为整型数值的初值

if(a>=10)

{

printf(”a>=10\n”); // \n为换行符

}

else

{

pintf(”a<10”)

}

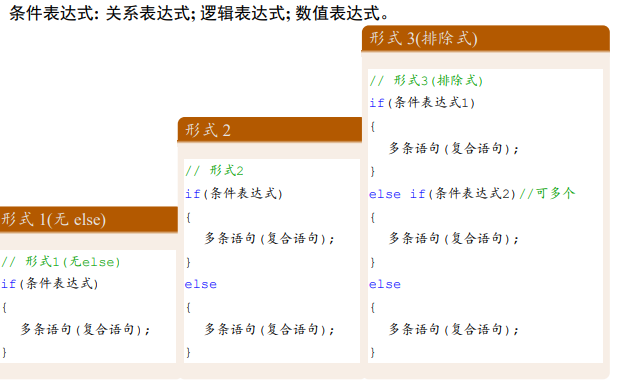
return 0; // 函数执行完毕返回函数值0

} // 函数结束标志

注意点：if条件表达式后不能加分号不能输出结果，后面要加括号，括号内为循环量。

“==”表示等于，“！=”表示不等于；

“&&”表示和，“||”表示或，“！”表示非。



关系运算符及其优先次序

算术运算符>关系运算符>赋值运算符

2.while(条件表达式)

#include<stdio.h>

int main()

{

int a=10;

while(a>=0)

{

printf(”a=%d\n”,a);

a--;

}

return 0;

}

while语句可以简单记为：只要当循环条件表达式为真，就执行循环体语句。

注意：1.while循环特点是先判断条件表达式，后执行循环体语句。

2. 循环体如果包含一个以上的语句，应该用花括号括起来，作为复合语句出现。

3. 不要忽略给i和sum赋初值，否则它们的值是不可预测的，结果显然不正确。

4. 在循环体中应有使循环趋向于结束的语句。

3.do，while(表达式);

#include<stdio.h>

int main()

{

int i=1,sum=0;

do

{

sum=sum+i;

i++;

}while(i <= 100);

printf(”sum=%d\n”,sum);

return 0;

}

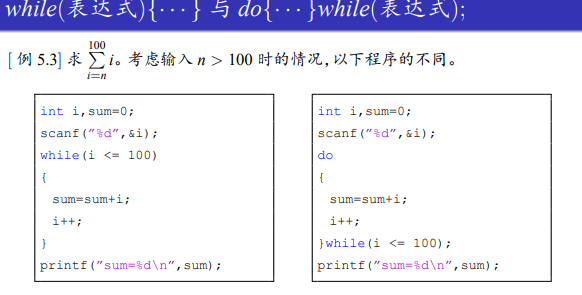
特点： 先无条件地执行循环体，然后判断循环条件是否成立。

while语句与do，while语句比较

1．在一般情况下，用while()语句和用do，while(); 语句处理同一问题时，若二者的循环体部分是一样的，那么结果也一样。

2．但是如果while后面的表达式一开始就为假 (0 值) ，两种循环的结果是不同的。

3.while语句while语句括号后面不能写分号，而do，while语句中while后面有分号。



4.for语句实现循环

for语句一般结构：

for（表达式1；表达式2；表达式3）

表达式1: 设置初始条件，只执行一次。可以为零个、一个或多个变量 (逗号隔开) 设置初值。

表达式2: 是循环条件表达式， 用来判定是否继续循环。在每 次执行循环体前先执行此表达式(包括第 1 次循环)，决定是否继续执行循环。

表达式3: 作为循环的调整，例如使循环变量增值，它是在执行完循环体后才进行的。

最常用for语句：

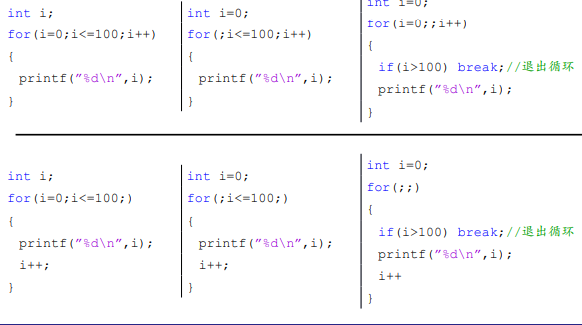
for（循环变量赋初值；循环条件；循环变量增量）

例如：

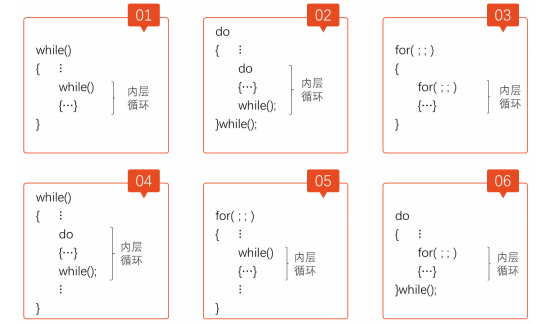
for（i=1；i<=100;i++）

sum=sum+i；

for语句省略表达式



5.循环的嵌套



6几种循环的比较

1.3种循环都可以用来处理同一问题，一般情况下它们可以互相代替。

2．在while 循环和 do…while 循环中，只在 while 后面的括号内指定循环条件， 因此为了使循环能正常结束，应在循环体中包含使循环趋于结束的语句 (如 i++ 等)。

3．for循环可以在表达式 3 中包含使循环趋于结束的操作，甚至可以将循环体 中的操作全部放到表达式 3 中 (逗号隔开)。因此 for 语句的功能更强，凡用 while 循环能完成的，用 for 循环都能实现。

4．用 while 和 do…while 循环时，循环变量初始化的操作应在 while 和 do… while 语句之前完成。而 for 语句可以在表达式 1 中实现循环变量的初始化。

5 .while 循环、do…while 循环和 for 循环都可以用 break 语句跳出循环，用 continue 语句结束本次循环。

四．分析上机练习注意点

1.运算注意事项：

1. 整数除以整数, 结果为整数。 4.0/3.0 结果是浮点数，4/3 结果是整数。

2. 化简公式会引起精度问题, 不要随意化简公式。

3. pow函数原型: double pow(double x,double y) 当形参数是整数时, 由于精度问题，不要使用此函数计算x y . 推荐使用循环语句, 易计算x y。

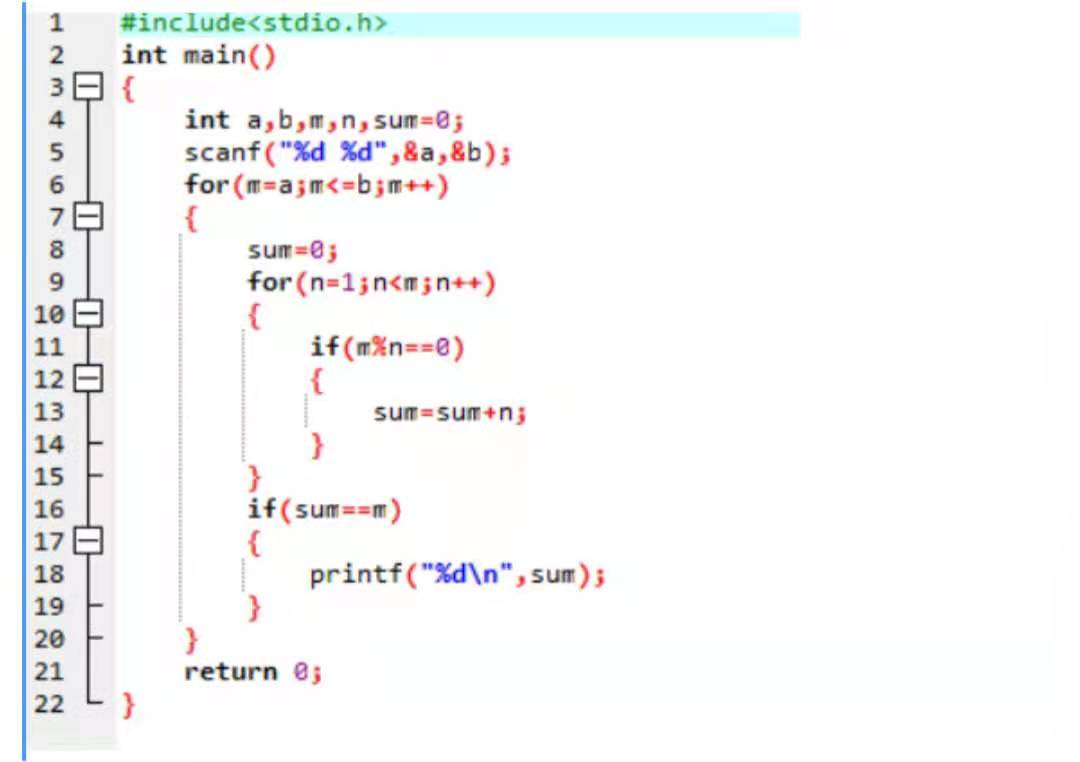
4. (整型数值与字符混合运算) 字符对应的 ASCII 编码参与整数运算, 其结果也是整数，注意’0’与0不同。

5. (精度问题) 32 位编译器: 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。

2.if语句的正确使用

例题2：

请写一个程序，给出指定整数范围 [a，b] 内的所有完数，0 < a < b < 10000。一个数如果恰好等于除它本身外的所有因子之和，这个数就称为 “完数”。



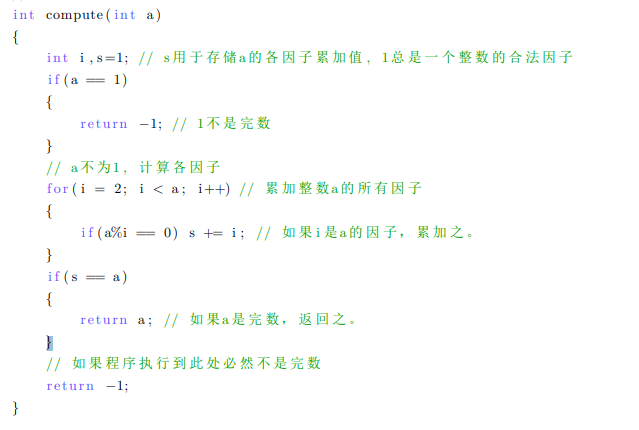
或者利用函数方法解决问题：

采用一重循环+调用函数方案

(1) 一重循环使整数i递增，函数compute调用，完成区间[ n1 , n2 ] 区 间的完数计算。

(2)定义函数compute , 判断整数参数是否是完数 , 如果是，返回它 , 否则返回−1。

// 定义函数compute , 判断整数参数a是否是完数 , 如果是，返回a , 否则返回−1。



(特别注意).切记：进入内层循环前, 相关变量的初始化问题。

3.角谷定理与欧几里得定理

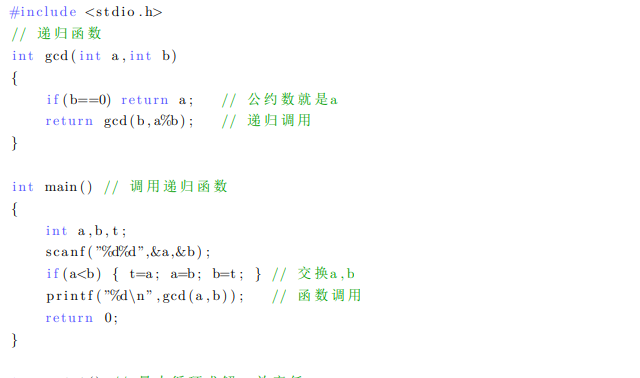
角谷定理：对于一个大于1的整数n，如果n是偶数，则n= n/2。如果n是奇数，则n = 3 \* n +1，反复操作后，n一定为 1。

欧几里得定理：a (大) ,b(小)的最大公约数:因为: a=mb+r , m=a/b ;r=a%b , ⇒a , b的公约数能整除和r .

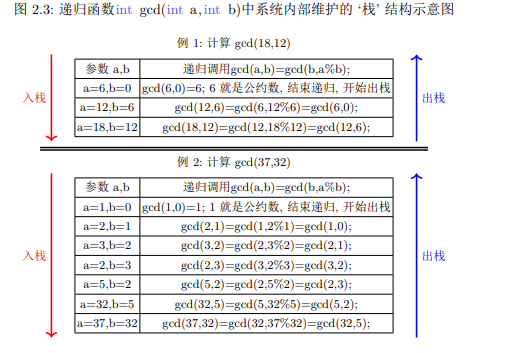
r=a%b , r为0 , 则b就是最大公约数否则迭代循环 , a=b，b=r , 直到余数为零 , 则分母是最大公约数。

4.递归函数

例：求最大公约数



递归函数入栈与出栈



后记：

在过去的四十年里，C语言已经成为世界上最流行、最重要的一种编程语言。 C语言是一种融合了控制特性的现代语言，而我们已发现在计算机科学的理论和实践中，控制特性是很重要的。其设计使得用户可以自然地采用自顶向下的规划，结构化的编程，以及模块化的设计。这种做法使得编写出的程序更可靠，更易懂。未来的学习中应该付出更多的精力来进行C语言的学习，不只停留在理论方面，多上机多敲代码，不能只靠理解书上的知识来提升自己，只有自己亲手打出来的代码再能深深的印在自己的脑子里，才能成为自己的东西。不要松懈，继续努力！

以上就是我的一些心得体会！

