

课程学习报告

课程名称：计算机导论与程序设计

班级：20200306

学号：20049200425

姓名：孙兴栋

**目录**

1. **课程学习目的-----------------------------------P1**
2. **知识点 -------------------------------------P2-P13**
3. **学习心得-------------------------------------P13**
4. **学习收获-------------------------------------P14**

**5.今后目标--------------------------------------P15**

1. **课程学习目的**

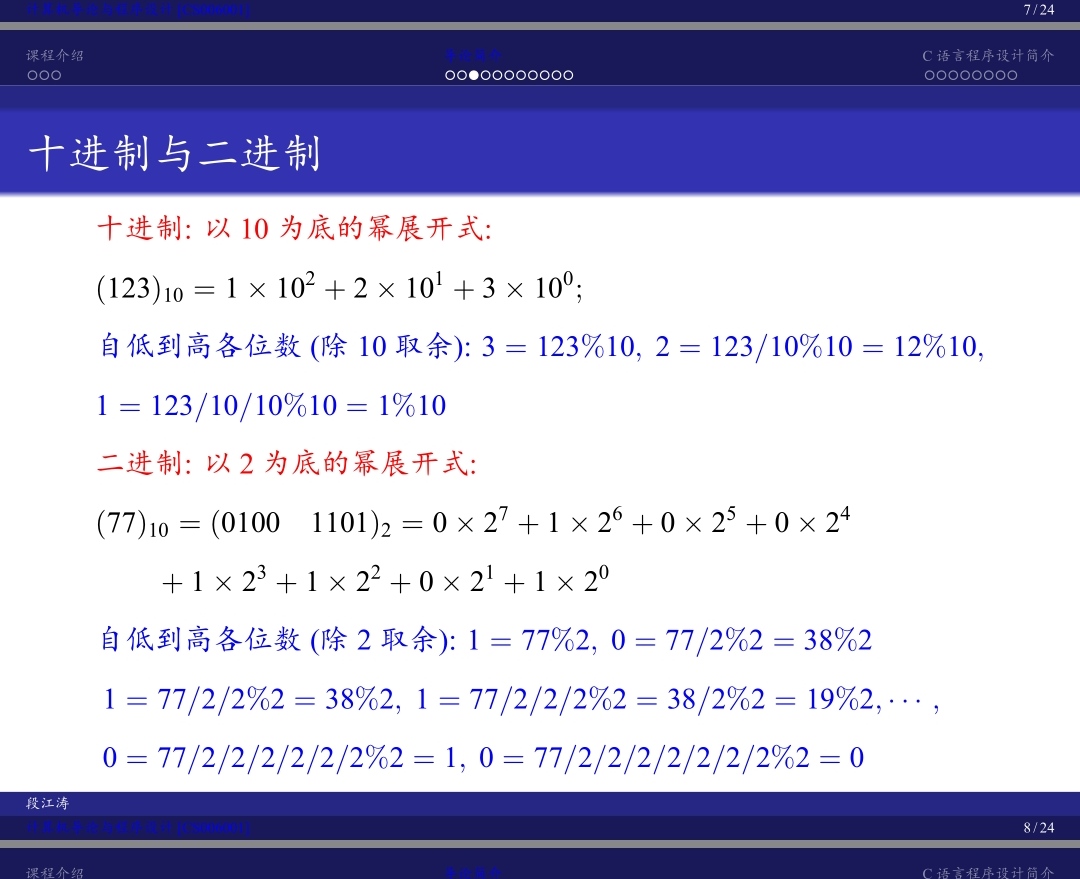
C语言的教学是学生根据课堂讲授内容，学生做相应的自主练习，消化课堂所讲解的内容；通过调试典型例题或习题积累调试C程序的经验；通过完成辅导教材中的编程题，逐渐培养学生的编程能力、用计算机解决实际问题的能力。

1. **知识点**
2. **计算机工作原理**：储存程序+程序控制

用二进制表示数据和指令，将程序存入储存器中，由中心控制器控制自动读取并执行。

外部储存存储的程序和所需数据发送到计算机内存，由CPU取出指令，分析指令，执行指令。

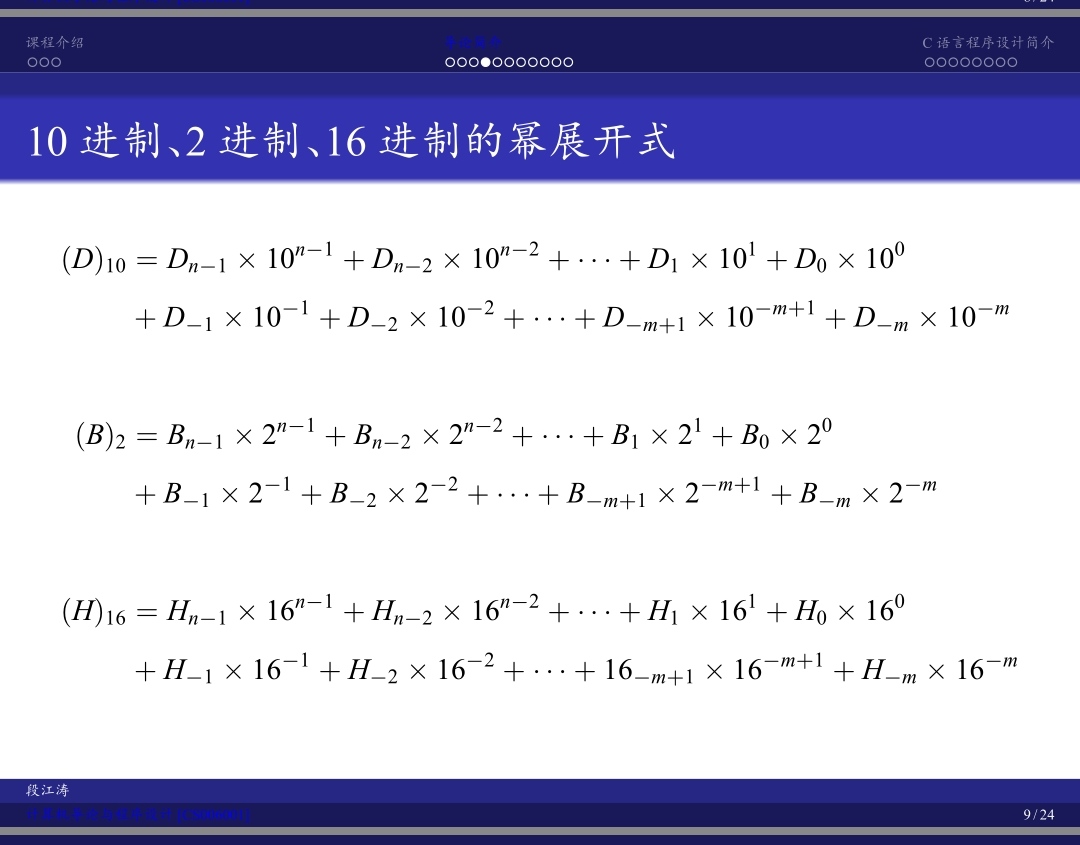
1. **十进制，二进制和十六进制**



**十进制是常见的进制表示**

**二进制是计算机常用的进制**

十进制，二进制和十六进制的幂展开式



十进制，二进制和十六进制的简单换算



1. **原码反码与补码**

对于正数 原码=反码=补码

对于负数，补码=反码+1

反码=符号位不变其他位按位取反，补码可逆即再对补码求补得到原码。

补码使减法统一为加法。

例：（+77）补+（-77）补=01001101+10110011=10000000

符号位是0的为正数，符号位是1的是负数。

1. **ASCII编码表**



1. **计算机程序**

指令：可以被计算机理解并执行的基本操作指令

程序：一组计算机能使别和执行的指令。一个特殊的指令序列用来完成一定的功能。

软件：与计算机系统操作有关的计算机程序，规程，规则，以及可能有的文件文档及数据

1. **第一个c语言程序**



**你好，世界！**

1. **常见的程序及运行c语言的步骤与方法**

.exe .c .cpp .pdf .ppt .doc .txt

开始-----编辑-----编译-----是否有错？------没错则连接------执行------结果正确？----正确则结束

------



1. **常用格式符与数据类型的对应关系**



不正确则重新开始

-----

有错则重新开始

int ------整数

float-----单精度浮点数

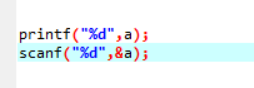
double------双精度浮点数

char------字母（后面的字母用‘’括起来）例：char ‘A’

sizeof------字节为

**9.printf与scanf语句**

printf是输出语句，scanf是输入语句



&为取地址符，scanf语句中必存在，否则无法运行。

scanf语句中不要指定精度。

getchar（）可作为消费回车键，也可作为输入字符，putchar作为输出字符。

运算之前要定义它的数据类型。

例：Int a，b //定义变量a，b位为整数，同类型的变量可以在一条语句中定义。

float f //定义变量f为单精度浮点数

double d //定义变量d为双精度浮点数

Char c //定义变量c为单个英文字母

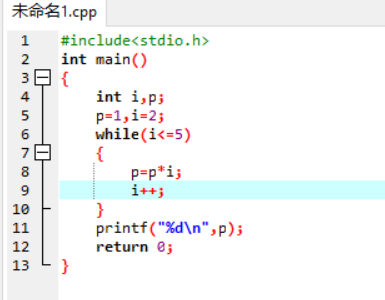
**10.while与if函数**

While（条件表达句）『表达式为真（非0）时执行语句』

If（条件表达句）『表达式为真（非0）时执行语句』

While，if函数之后不加分号不能输出结果，后面要加花括号，括号内均为循环量，若不加括号，只会进行第一项。

例：求5！的c语言程序



**11.输出字符的ASCII码**

Char y=‘X’

Printf（“%c”，x）； 得字符x

Printf（“%d”，x）； 得字符变量的ASCII码 10进制

Printf（“%x”，x）； 得字符变量的ASCII码 16进制

Printf（“%c”，x）； 得字符变量的ASCII码 8进制

unsigned int-----无符号整数，%u----无符号位输出。

例：unsigned int f=0xFFFFFFFB（0x后为数字的十六进制表示）

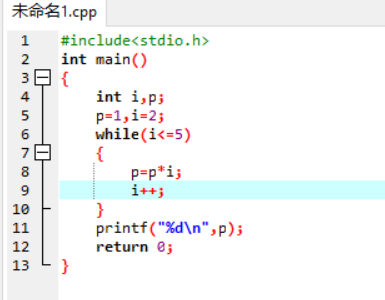
Printf（“%u”，f）

%在引号之外是取余的意思。

**12.数学库函数**

math.h

当需要使用数学函数时，要在最前面添加数学库函数，例如使用对数，绝对值，n次方时。



1. **优先级**

算术运算符优先于关系运算符优先于赋值运算符

=是赋值 ==是判断是否相等

!a是非a，&&是和，||是或

在定义一个常数时，可以在最前方定义，格式为

#define PI 3.1415926

即为定义Π的值。

在表示10<=a<=100时，不用if（10<=a<=100）

而用if(a>=10&&a<=100)。

1. **switch语句和break**

Switch(x)

{

Case 0;

Printf(“X=0\n”)

Case 1;

Printf(“X=1\n”)

}

Break 用在printf（）之后，用来使执行中断。

注意：case 和default后面都加：

若每一个case都不满足，则用default。

Case只能用于等于整数。

1. **for函数和do while函数以及continue的用法**

For函数的格式 for（ ； ；）

{

}

第一个空填写初始表达式，第二个空填条件表达式，第三个空填循环变量增量。

循环变量增量例如i++用于case函数里面的变量增加。

Do while函数的格式 do

{

}while（）

条件写在while后面的括号中。

Continue用于终止循环，开始下一轮的循环。

1. **运算注意事项**

1.整数除以整数，结果为整数。例：3/4=0，没有四舍五入，直接取整，3.0/4.0结果是浮点数，才会是0.75。

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）

当型参数是整数时，由于精度问题，不要使用此函数计算x的y次幂，推荐使用循环语句，易计算x的y次幂。如果必要，可以使用函数int mypow（int x，int y）。

1. 整型数值与字符混合运算时，字符对应的ASCII码参与整数运算，其结果也是整数。注意‘0’与0不同。输入0，0，则a=0，b=‘0’，变量a的值是整数0，变量b的值时字符‘0’所对应的ASCII编码，即整数48.
2. 输入输出格式转换符不对应。例：int a;float b;char c;double d;

Scanf(“%d”,a) 遗忘变量前的取地址符&

Scanf(“%d\n”,&a) 多余的换行符，导致不能运行。

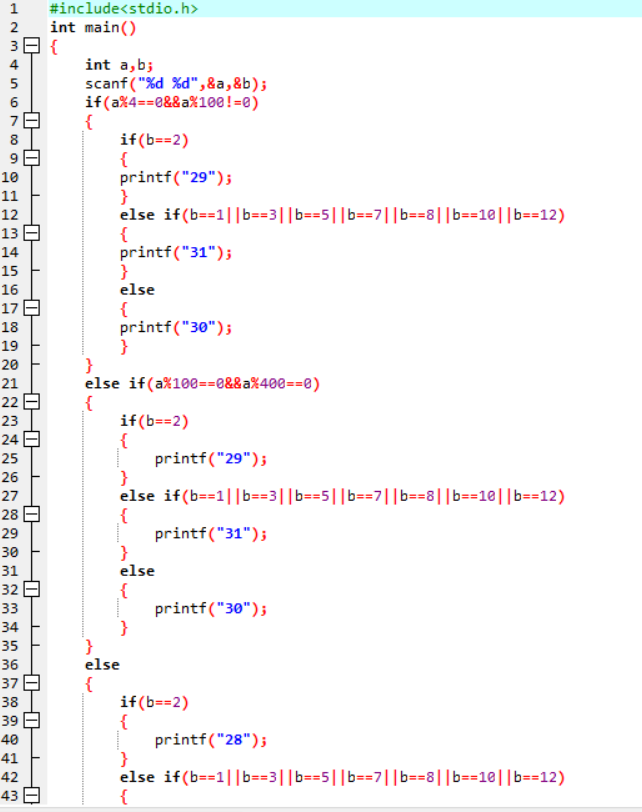
1. if，while，for，函数后不能加；否则不能执行。
2. 表示范围错误 例如：if（110<=a<=210）错误

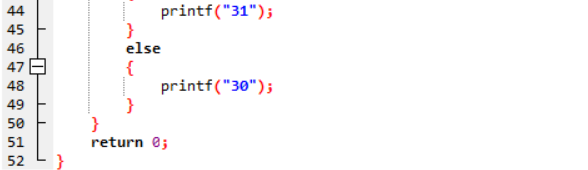
If（100<=a&&a<=210）正确

1. 使用printf（）语句，追踪程序执行细节，查找出错原因。
2. 对于条件结构，循环结构，首先书写整体结构，再添加细节，避免低级错误。
3. **if函数的结合运用与格式**

例题1：计算某月的天数

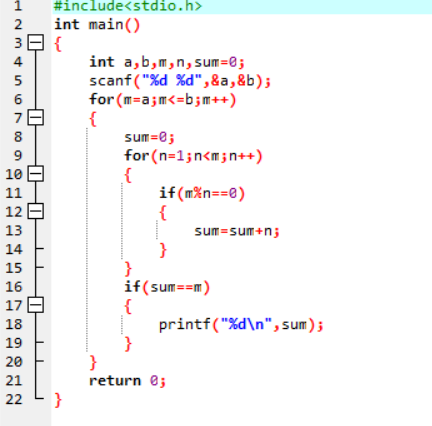
每年的1，3，5，7，8，10，12月有31天，4，6，9，11月有30天，闰年2月29天，平年2月28天，给定年份和月份求该月的天数：





例题2：

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，采用定义函数的办法：

//定义函数compute判断整数参数a是否为完数，如果是，返回a，否则返回-1

Int compute（int a）

{

Int i,s=1; //s用于储存a的各因子累加值，1总是一个整数的合法因子

Int ret=-1; //用于返回值，默认为-1

for(i=2;i<a;i++) //累加整数a的所有因子

{

If(a%i==0) s+=i; //如果i是a的因子，累加之

}

If(s==a&&a!=1) //如果a是完数，返回值是本身，1不是完数

{

ret =a

}

Else //a不是完数

{

ret =-1

}

Return ret;

}

Int main()

{

Int i,n1,n2;

Scanf(“%d%d”,&n1,&n2);

For(i=n1;i<=n2;i++) //调用函数compute，完成区间[a,b]的完数计算

{

If(compute(i)!=1) //如果a是完数，输出；测试函数compute的调用

Printf(“%d\n”,i);

}

Return 0;

}

注意事项：切记进入循环之前，相关变量的初始化问题。

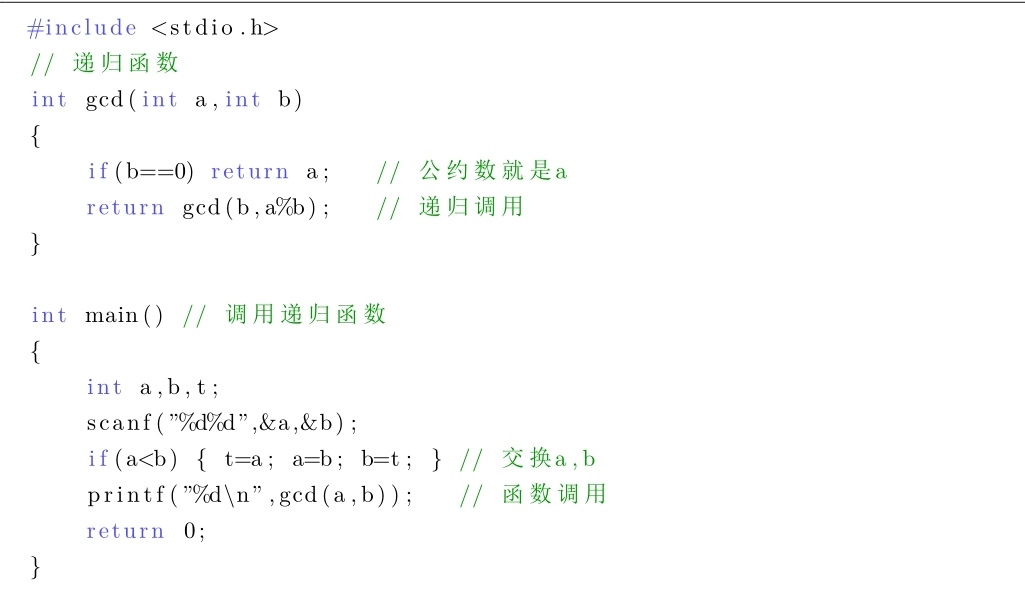
void表示无返回类型的函数，此函数不返回任何数据。

1. **欧几里得原理和角谷定理**

欧几里得原理：a(大)，b（小）的最大公约数：因为a=mb+r，m=a/b；r=a%b，即a，b的公约数能整除b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0，则分母就是最大公约数。

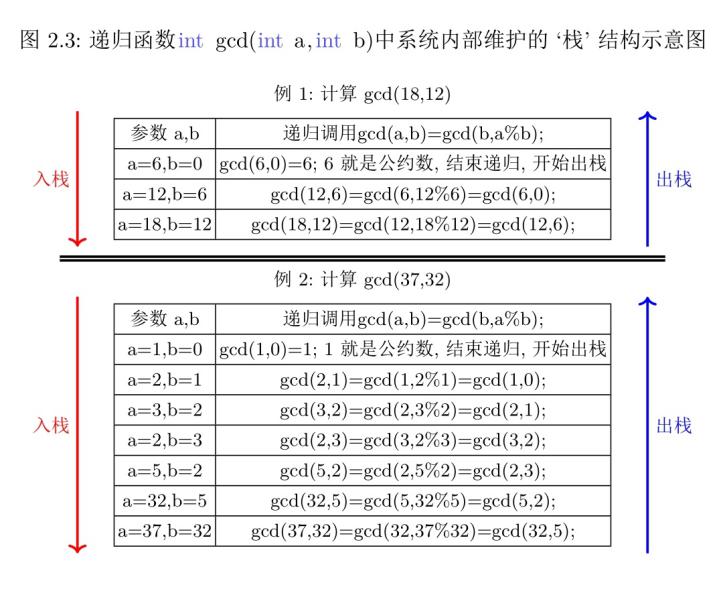
角谷定理：对于一个大于一的整数n，如果n是偶数，则n=n/2。如果n是偶数，则n=3\*n+1，反复操作后，n一定为一。

1. **递归函数**

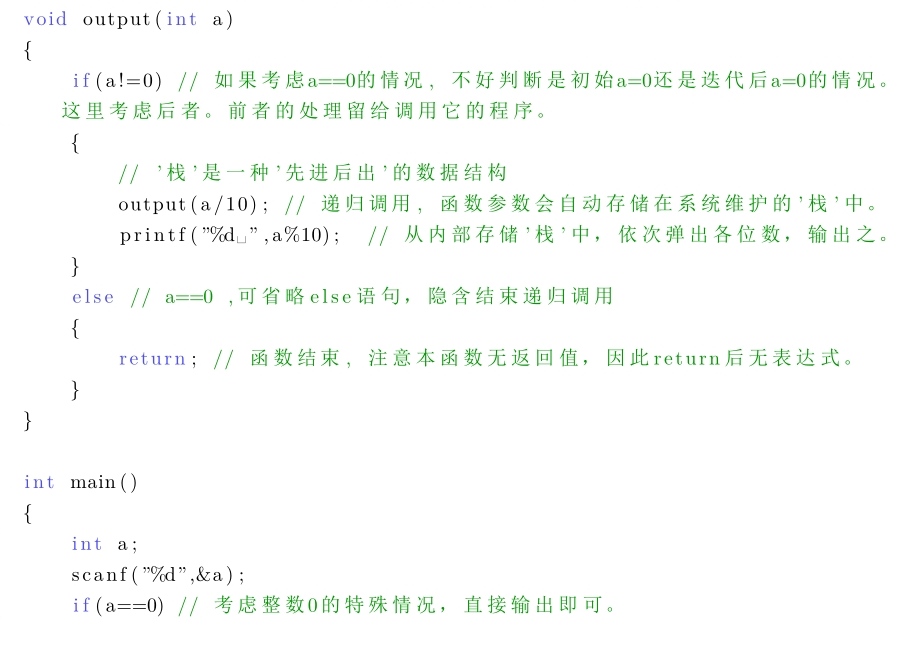
例：求最大公约数

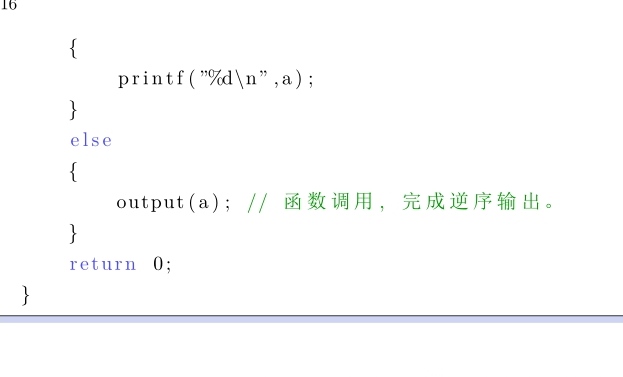
先定义一个函数，在函数执行过程中进行递归调用，就形成了一个递归函数。

**递归函数的入栈与出栈**

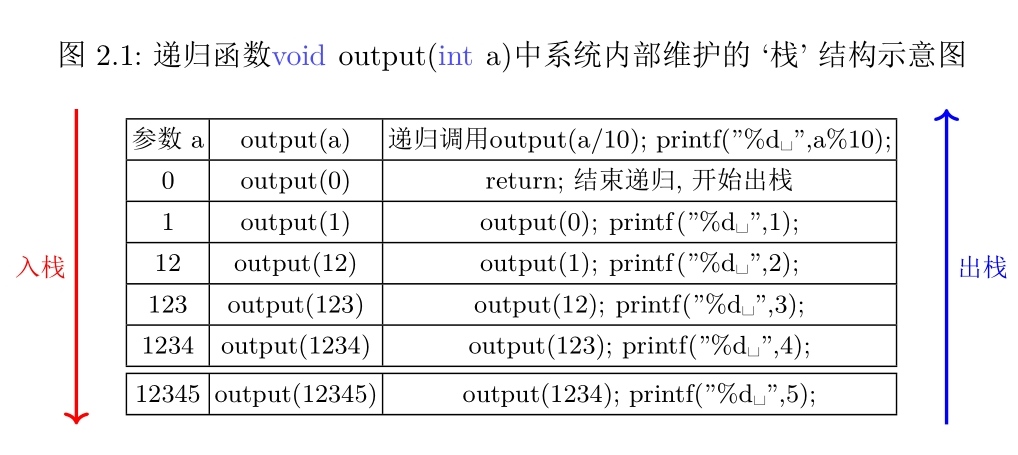


例：数位输出





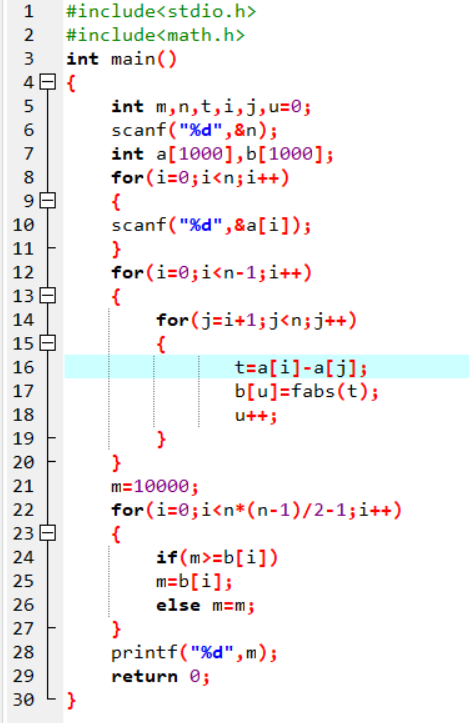
递归函数的入栈与出栈



**20.数组的应用**

数组指一个数组内含有多个数值，定义为a【100】则表示该数组内右至多100个数字。

例题：输出两数之差绝对值最小的数 数组内第一个数为第零项。



**三．学习心得**

学习无捷径！对于学习编程而言，我们现在的付出将来都是有回报的。但是，学习C语言也需要方法。学习C语言不能半途而废，需要找到正确的学习方法！如果在学习的过程中四处碰壁，兴趣和自信心逐渐被消耗殆尽，那么学习C语言将是一件很痛苦的事。事实编程是一件很好玩、很有趣、很有意思也很有前途的事情！学习C语言有很多好的方法。我认为分为以下几种方面。

**1.分清主次**

学习C语言最忌讳的就是不分主次，这是很多人都会犯的错误，我们刚开始学习的时候只需要将那些最重要的、最核心的学会就可以。先将最精髓的东西提炼出来，再将整个C语言学一遍，从全局上把握C语言。对于那些次要的，有需要再学，没有需要也可以不学。

#### 2.一定要多上机，多“敲”代码

编程是一门实践性的学科，绝对不是理论。如果不动手“敲”代码的话，永远都学不会编程。很多问题只有在“敲代码”的时候才能发现，才会有更加深刻的体会、领悟和理解。而不是靠死记硬背书中的注意点，那样真的很痛苦。在敲代码的工程中，我们往往能收获很多。

#### 3要学会记笔记

正如老师所说，编程需要不断地积累。我们一定要学会模仿别人优秀的代码、优秀的算法，然后将它记下来。但是我们的记忆能力是有限的，时间长了难免会遗忘，所以一定要学会记笔记。一有心得、体会、感悟就写下来，这些都是很珍贵的。我们在记笔记的时候，应该写在纸上，才能更好的记忆。

**四．未来学习目标**

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！

**完。**

****