C语言学习报告

基本语法

#include<stdio.h> //standard input/output 编译预处理指令

int main() //主函数

{} // 函数开始标志与函数结束标志

printf(“ ’’) //输出一行信息

return 0 //函数执行完毕返回函数值0

; //每一条语句结束后必加上 ;

格式符: %d,%f,%c,%lf,%.2f,%.2lf,%x,%ld

对应的数据类型 int,float,char,double,float,double,int,long int

%.2f //保留两位小数

%.3f //保留三位小数

int a,b; //定义变量a,b为整型数值

float c; //定义变量c为单精度浮点数

char c; //定义变量c为单个英文字母

double d; //定义变量d为双精度浮点数

定义变量时可以给变量初值如：int a=10，float b=10.2，且同一类型变量可以在同一行中定义，如int a，b，c，d

char c=’A’ //给变量c字符A时需要加上单引号

\n //换行符

= //赋值

== //等于

！= //不等于

& //取地址符

getchar //字符输入函数，消费上一步的回车键

putchar //字符输出函数

scanf() //输入函数，变量前要加上取地址符&

scanf(“%d”,&a) //输入整型变量a的值，注意在a前面加上取地址符&

char a='b';

a=getchar() ; //从键盘输入一个字符送给字符变量a

putchar(a); //将变量a的值输出

#define PI 3.14 // 定义常量pi=3.14，没有分号结尾

char a='a',b; //定义变量a，b，把a赋给变量a

b=a-32; //把a转换成大写字母A

printf("%x",b); //输出十六进制的A的ASCII编码

int a=65;

printf("%c",a); //将65转换成A输出

int a=65,b=10,c;

c=a/b; //整数除以整数只能等于整数

printf("%d",c); //c=6而不是c=6.5

int a=65,b=10;

float c;

c=a/1.0/b; // a/1.0可以得到预期结果

printf("%.1f",c); //c=6.5

i++,i--; //i=i+1,i=i-1

int a,b=10;

a=++b; //先加1后赋值，即b=10+1=11,a=b

a=b++; //先赋值后加1，a=b=10,b=10+1=11

#include<math.h>//数学库函数调用

常见函数表达式

abs(x) //求整数x的绝对值

fabs(x) //求整数x的绝对值

sqrt(x) //求整数x的算术平方根

pow(x,y) //求整数x的y次方

log(x) //求lnx

log10 (x) //求log10x

选择结构与条件判断

if 语句, 用来实现两个分支 的选择结构； switch 语句, 用来实现多分支的选择结构

if() //条件表达式，注意括号后不加分号

int a,b,t;

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

if(a>b) //判断输入a，b的大小

{

t=a;a=b;b=t; //通过第三个变量t完成a，b值互换

}

printf("%d %d",a,b);//从小到大输出

1. if(a>b)

else //当a，b满足第一个条件就执行第一个，不满足第一个时执行第二个

if(a>b)//判断输入a，b的大小

{

t=a;a=b;b=t; //若a>b,执行赋值语句

}

else

{

a=10,b=10; //若a<=b,执行a=b=10这条语句

}

2. if(a>b)

{

t=a;a=b;b=t;// a>b,执行第一个条件语句

}

else if(a<b)

{

a=10,b=10; // a<b,执行第二个条件语句

}

else

{

a=0,b=0; //a=b,执行第三个条件语句

}

数学语句与与c语言语句表达的不同

如数学语句表示0<=a<=10,在c语言中应该表示为

if（0<=a&&a<=10）// &&表示且的意思

若写成 if（0<=a<=10），则9与50都是满足这个条件的

&& // 且

|| //或

Switch与if条件语句的区别

int a,b,c; int a,b,c;

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

if(a>=b) c=a/b;

{ switch(c)

a=0,b=0; {

} case 0: a=10,b=10;break case 1: a=0,b=0;break；

else(a<b) }

{

a=10,b=10;

}

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

break //终止整个循环

循环结构程序

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。

1. while(a!=0) //判断条件当a！=0时循环，括号后无分号,可简写为while（a）

{

b++; //每循环一次b加1，每循环一次c都等于a对10取余，a每循环一次等于前一次的十分之一，当a=0时循环终止

c=a%10;

a=a/10;

}

1. for(初始化表达式如i=0;条件表达式如i<n;循环变量增量如i++)

{}

1. do

{

b++, c=a%10;a=a/10;

}while(a!=0)

注意：使用循环时一定要养成给循环变量i初值的习惯，否则i因为值不确定会造成结果错误，在使用多个循环及内外层循环时，先写好循环的结构，然后再写具体程序，防止括号乱打或忘打

break与continue终止循环

1. break 语句只能用于循环语句和 switch 语句之中，而不能单独使用
2. continue 语句只结束本次循环，而非终止整个循环。break 语句结束整个循环，不再判断执行循环的条件是否成立。

数组

a[100] // 表示100个数的集合

a[0],a[1],a[2],a[3],a[4],a[5],[6]一直到a[99]表示每一个数，没有a[n]

这种表示方法，也没有a[100]这个数，因为是从零开始的

sacnf（”%d”,&a[i]; // 同样加取地址符

在不能判断集合有多少元素时通常往最大值计

易错点

1. 每条语句结束后忘记分号，在循环时如if（）不应该加分号的地方加分号
2. 把赋值符号=与等于==误用
3. 输入符号时因为输入了中文符号导致出错
4. Scanf用来输入时忘记取地址符&
5. 输出格式没有对应定义变量类型
6. 整数除以整数只能等于整数，没有加.0进行除法运算导致结果中小数点后面的数字被省去而出错
7. 对于运算顺序不清而出错，如a/b/c与a/c/b,假设其中有余数就会出现精度损失（小数点后面的数字被省去而出错）
8. 运用多个if和for循环时括号的缺少或多添
9. 循环时忘记定义i的初值
10. 内外层循环分不清看不明白导致错误
11. 没有严格遵循计算机的逻辑因为个别表达式先后顺序导致出错
12. 数学思维与计算机语言思维不同未能及时转变，如0<=a<=100

C语言学习心得

作为一门全新的以前从未接触的新学科，开始学时总觉得晦涩难懂，上课也听不懂老师讲什么，但在第一次上机后我逐渐明白这是一门更重上机操作的学科。在上机时可以把上课听不明白的东西用于实践并且助力于对理论知识的理解。那其实这也是一门多练的科目，练的多了自然就明白并贯通了理论知识。反之，理论知识再好不多上机练习也很难真正明白到底你认为的是否就是对的。这也是一门很重逻辑的学科有助于培养逻辑推理能力。反正，我相信只要努力练习就可以学好，他就像认汉字一样从开始的一字不识到每天的日积月累到后来的融会贯通只是一个不断熟练的过程。