计算机导论学习报告

在学习了该课程半学期后，对C语言有了部分的了解，在此作学习总结的报告。

* 循环语句

循环语句是课程中首个学习到的难点，有while、for、switch三种形式，while和for区别较小，switch与for和while区别较大。

先说说switch语句，应用在某个值的选取上用switch（c）,来区别不同情况下的C值所进行不同的命令。格式大致如下：

switch(c)

{

case 1:\*\*\*\*\*；break；

case 2:\*\*\*\*\*；break；

case 3:\*\*\*\*\*；break；

……

}

要注意的是 1：在每个case语句之后都要有一个break语句来结束，否则会接着进行下面的命令导致错误。2：case语句是有冒号的不能漏 3：case是不能进行范围式的选择的，c只能是其中某一值。

接下来是while语句，与switch语句相比较为简单，限制的因素比较少比较简单，有两种写法。如下：

While(条件。。。。)

{

\*\*\*\*\*\*

}

也有：

Do

{

\*\*\*\*\*\*

}while(条件。。。。)

两种写法，当满足条件时执行，但是两种写法存在着细微的区别，第一种时满足条件便执行，第二种是先执行一次，如果满足条件便进入循环在执行一次，如此往复，直到不满足条件便跳出循环。但是通常也配合break和if或switch来进行跳出循环的操作。如:

While(条件。。。。)

{

\*\*\*\*\*\*

If(条件2.。。。。。)break；

\*\*\*\*\*\*

}

最后是for循环，规则更明确，更清晰在修改差错的时候更加容易，非常推荐。也会配合着continue和break来跳出循环，也会多个for叠加来实现递归和各种比较复杂的操作。但是for的运用也有规则，在括号中，从前往后依次是赋值初始变量，循环条件，每次循环的改变量。还有对于多个for循环的叠加和累加的值，要进行初始化操作。例子：

#include <stdio.h>

int shuweihe(int n)

{

int i,s=0;

for(i=10;n;n=n/10)

{

s=n%10+s;

}

return s;

}

int main()

{

int n;

scanf("%d",&n);

printf("%d",shuweihe(n));

return 0;

}

* 逻辑函数

逻辑函数有三种，分别为&&、||、？=三种情况，配合大于、小于和等于多用在if条件句中来达到选择的目的。例子：

int a=5,b=10,c=0; //以int为例

if(!a) // 逻辑非(NOT), a是非0, 所以!a的值是0

{ ... }

if(a && b) // 逻辑与(AND), a,b均为非0, 所以(a && b)的值为1

{ ... }

if(a || c) // 逻辑或(OR), a,c之一是非0, 即为真

{ ... }

* 函数调用

函数调用在写较为复杂的问题时很常见的手法，他能极大的简化函数，方便差错，函数调用分为两种，分别是有返回值的函数和无返回值的函数。如下

先说有返回值的函数，例如：

#include<stdio.h>

// 自定义整数pow函数

int intpow(int x, int y)

{

int i,p=1;

if(x==0) return 0;

for(i=0;i<y;i++)p=p\*x;

return p;

}

int main()

{

int x=2,y=3;

printf(”%d,%d\n”,intpow(10,x),intpow(5,y));

// 100,125

printf(”%d,%d\n”,intpow(10,2),intpow(5,3));

// 100,125

printf(”%d,%d\n”,intpow(0,2),intpow(0,0));

// 0,1

return 0;

}

如此调用在多次利用到同一个计算结构的时候就可以很好的简化函数。

另一个便是无返回值的函数。如：

void output(double a)

{

printf(”%lf\n”, sqrt(a)); // 函数

中可调用别的函数

}

调用这样的函数便可以执行一些不需要返回值的反复操作，但是要注意的是要用void开头，并且没有return。

* 反思

以上便是在其中前所学习的重点知识了，在接下来的半学期中还有许多的知识等待着我们去发掘，但是稳定的复习和对所学知识的应用和实践仍旧是这门课所必须的，希望在接下来的半学期中继续加油继续努力。