**(3) 循环**

while [ 表达式 ]

do

命令

done

只有在表达式为真的时候，才执行命令,如果为假，循环结束。

while true

do

…….

done

**Practice 8:**

**计算1 + 2 + ...... + 100**

for 变量 in 单词表

do

命令

done

循环的次数由单词表中单词的个数来决定，变量每次从单词表取一个单词，然后执行循环体，直到将所有的单词全部取完。

举例：

i=1

weekdays="Mon Tue Wed Thu Fri Sat Sun"

for day in $weekdays

do

echo "Weekday $((i++)) : $day"

done

如果 "in 单词表"省略，此时变量从位置参数中取单词：

for 变量

do

命令

done

**Example**：

num=1

for i

do

echo "The ${num}th argument is: $i"

((num++))

done

**Practice 9:**

**输入一个路径，并判断该路径是否合法（是否存在该路径），如果合法，则输出该路径下的普通文件以及其它文件的个数。**

**三 shell 函数**

1. **函数定义**

**function 函数名()**

{

命令

...

return 数字值

}

**2. 函数调用**

第一种调用方法:

**函数名 参数1 参数2 ...**

**通过 $? (上一个函数或者命令返回的结果)，获得函数的返回值**。函数本身要在屏幕上输出结果。

(**适合返回的是一个整数，并且只能够返回0-255之间的整数**)。

**在函数内部通过位置变量($1, $2, $3 ...)来获得传递的参数**。

第二种调用方法:

**value=`函数名 参数1 参数2 ...`**

**将函数的标准输出返回给一个变量（使用反向单引号）**，**函数本身不再在屏幕上输出结果。**相当于是把函数的输出赋值给一个变量，而并没有使用函数本身的返回值。

**无论函数定义中有没有返回值，这种调用方式函数返回值始终是0**。

(**适合函数输出/返回的是一个字符串**)

注意:在shell中变量默认的作用域是整个文件的，即使是在函数内部定义的函数，其它函数也可以使用。如果需要变量只在函数内部有效，需要用local修饰变量。

函数的载入

如果函数在另外一个文件中，我们该怎么调用它呢？

比如 show 函数写在了function.sh里面了，我们可以用 source 命令：

source function.sh

这样就可以调用了。

**Practice 10:**

**设计一个脚本程序完成以下需求：**

**1. 设计一个函数，通过传递的用户名来寻找用户在/etc/passwd中的行号(存在:返回行号，不存在:返回0)。** （提示：如果查找的行号结果为空字符串，则表明未找到该用户）

**2. 设计一个函数，通过传递的行号，输出这一行对应/etc/passwd文件中用户信息如下**

**The user info: NAME:UID:GID**

**3. (主函数)循环读取用户输入的用户名，然后分别调用设计的函数，输出用户的信息，如果用户不存在，则告诉输入者这个用户在系统中不存在，如果输入的是"quit", 则结束。**