

c语言第四天

exit()函数

程序的任何地方运行到这个函数，程序立即结束

需要传入

1 成功的情况下退出

-1 失败的情况下退出

标记位

```
int flag = 1;
通过给flag不同的取值，从而可以判断程序执行的状况，例如：
if(成立)
{

}

}else if(){
    flag = 0;
}else{
    flag = 2;
}
```

下一步的操作，基于上一个if判断执行的状况而进行，则可通过判断flag取值的方式得到上面的if语句执行了哪一部分

switch语句

```
switch(表达式){
    case 常量表达式1:
        语句1;
        break;
    case 常量表达式2:
        语句2;
        break;
    case 常量表达式n:
        语句n;
        break;
```

```
default:
    语句n+1;
}
```

当表达式的值与某个常量表达式的值相等时，即执行其后的语句，然后不再进行判断，继续执行后面所有case后的语句。如表达式的值与所有case后的常量表达式均不相同，则执行default后的语句。

if与switch的适用情况

if() 分支情况比较少
switch() 分支情况比较多

生成随机数：

```
srand((unsigned)time(NULL));
```

设置随机数种子，如果没有，则每次的随机数是一样的。而且在整个程序中，只需要出现一次就可以了

```
rand()%5    生成0 1 2 3 4 中的随机数
```

```
rand()%10 + 90 生成90---99之间的随机数
```

for循环

```
for(循环变量赋初值;循环条件;循环变量增值)
{
    循环语句;
}
```

执行顺序：1、初始语句 2、执行条件是否符合？ 3、循环体 4、增加增量

for循环的循环体如果只有一条语句，for循环的大括号可以省略

(1)for语句的一般形式中的"表达式1"可以省略,此时应在for语句之前给循环变量赋初值.注意省略表达式1时,其后的分号不能省略.如for(i<=100;i++){....};

(2)如果表达式2省略 即不判断循环条件,循环无终止地循环下去,也就是认为表达式2始终为真.

(3)表达式3也可以省略,但此时程序设计者应另外设法保证循环能正常结束.

(4)可以省略表达式1和表达式3,只有表达式2 即只给循环条件.

(5)三个表达式都可以省略,如:

for(;;)语句

相当于while(1)语句

即不设初值 不判断条件(认为表达式2为真值)循环变量不增值,无终止的执行循环体.

(6)表达式1也可以是设置循环变量初值的赋值表达式,也可以是与循环变量无关的其他表达式.

while循环

while循环和**for**循环其实大同小异

格式:

```
while(条件)
{
    循环体;
}
```

只要给定的条件为真, while 循环语句会重复执行循环体

作业:

1、编写程序

任意输入10个数, 分别计算输出其中的正数和负数之和。

2、编写程序

计算1---100以内的所有含6的数的和。

3、预习嵌套循环