

第五章 循环结构

循环型程序设计
辅助控制语句

➤ 5.1 工作场景导入

➤ 5.2 while型循环

➤ 5.3 do-while型循环

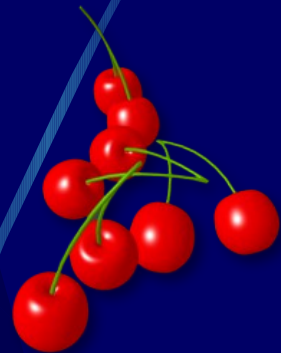
➤ 5.4 for型循环

➤ 5.5 其他控制语句

➤ 5.6 回到工作场景

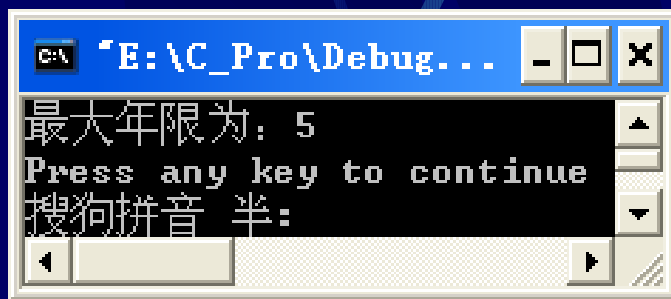


5.1 工作场景导入



● 【工作场景】

- 北京某毕业生需要贷款 100 万买房，贷款的银行年利率为 7%，由于此同学工资水平有限，欲尽量拖延还款的结束日期，但其认为还款总金额不可超过 150 万。此外，银行规定，还款期限不得超过 30 年。试为此人编程求出满足他的要求的最大还款期限，得到如下图所示的输出结果（精确到年）。



```
E:\C_Pro\Debug...
最大年限为: 5
Press any key to continue
搜狗拼音 半:
```

● 【引导问题】

- (1) 如何通过循环控制结构来解决此问题？
- (2) 运用何种循环结构来具体描述此场景？
- (3) 循环结构 (while、do-while 或者 for 短语) 具体如何使用？

5.2 循环型程序设计

- 概述

C 语言可实现循环的语句：

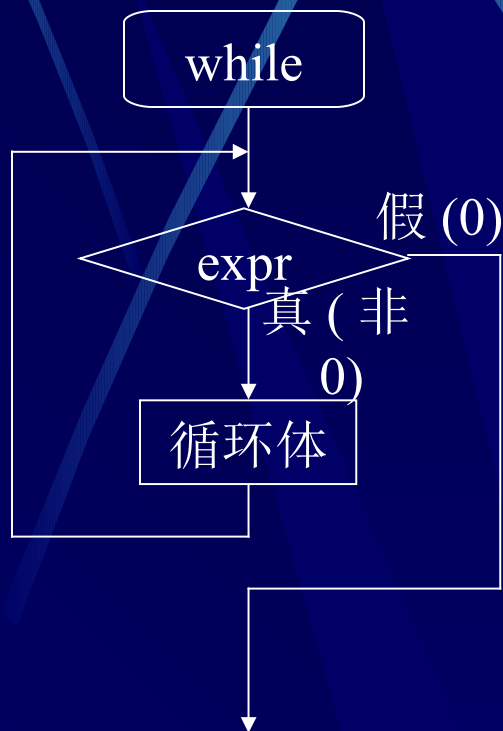
- 用 goto 和 if 构成循环（已不再使用）
- while 语句
- do ~ while 语句
- for 语句

while 语句

❖一般形式：

```
while( 表达式 )  
    循环体语句 ;
```

❖执行流程：

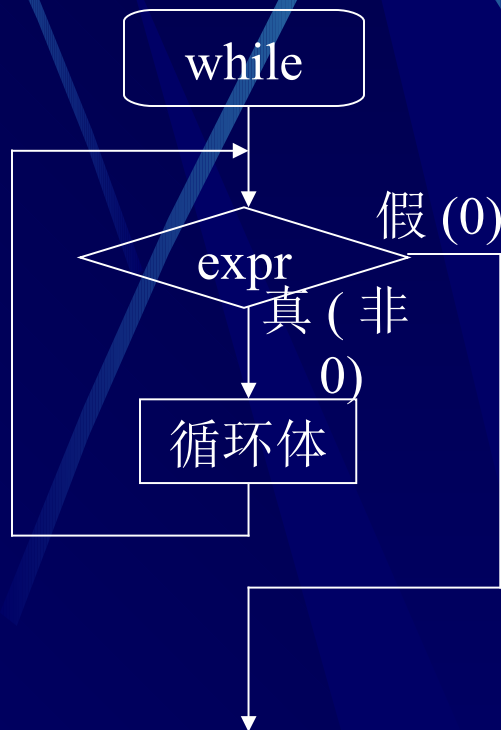


while 语句形式和执行流程

❖一般形式：

```
while( 表达式 )  
    循环体语句 ;
```

❖执行流程：



While 语句特点和说明

❖特点：先判断表达式，后执行循环体

❖说明：

- 循环体有可能一次也不执行
- 循环体可为任意类型语句
- 下列情况，退出 while 循环
 - ◆条件表达式不成立（为零）
 - ◆循环体内遇 break, return, goto
- 无限循环：

```
while(1)
    循环体；
```

例子 用 while 循环求 $\sum_{n=1}^{100} n$

```
/*ch5_1.c*/  
#include <stdio.h>  
main()  
{ int i,sum=0;  
  i=1;  
  while(i<=100)  
  { sum=sum+i;  
    ++;  
  }  
  printf("%d",sum);  
}
```

循环初值

循环条件

循环变量增值

循环终值

循环体

例 显示 1~10 的平方

```
/*ch5_2.c*/  
#include <stdio.h>  
main()  
{ int i=1;  
  while(i<=10)  
  { printf("%d*%d=%d\n",i,i,i*i);  
    i++;  
  }  
}
```

运行结果：

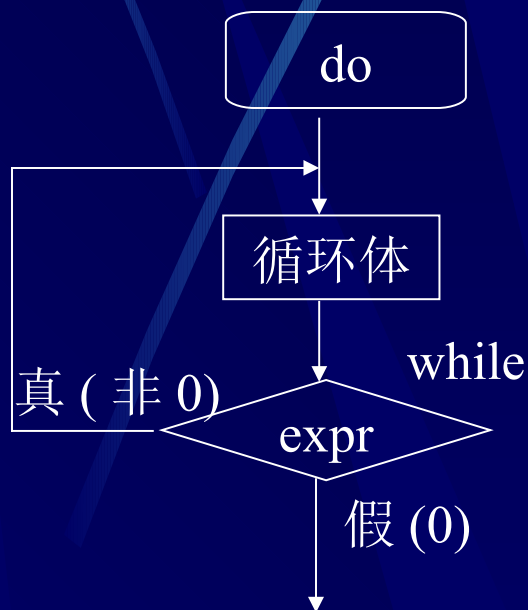
```
1*1=1  
2*2=4  
3*3=9  
4*4=16  
5*5=25  
6*6=36  
7*7=49  
8*8=64  
9*9=81  
10*10=100
```

do~while 语句

❖ 一般形式：

```
do  
    循环体语句 ;  
while( 表达式 );
```

❖ 执行流程：

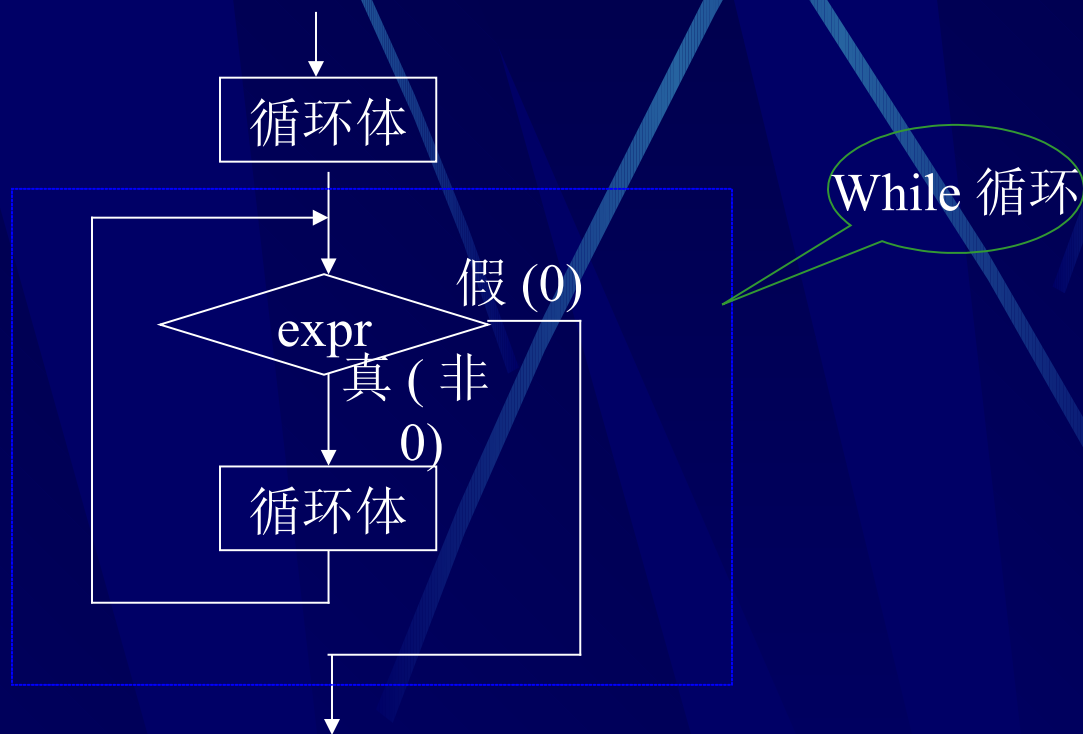


do~while 语句特点和说明

❖特点：先执行循环体，后判断表达式

❖说明：

- 至少执行一次循环体
- do~while 可转化成 while 结构

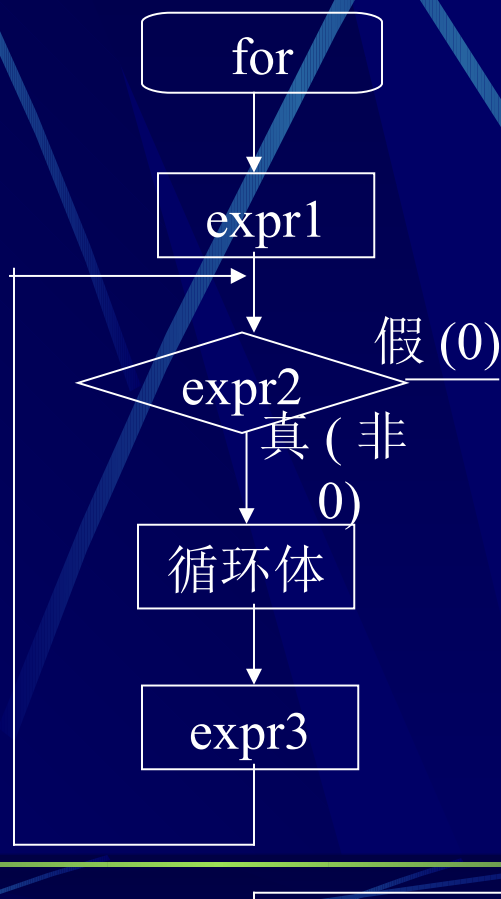


❖一般形式：

for 语句

```
for([expr1];[expr2];[expr3])  
    循环体语句;
```

❖执行流程：



❖：for 语句一般应用形式

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

❖说明：

- for 语句中 expr1, expr2, expr3 类型任意，都可省略，但分号；不可省
- 无限循环：for(;;)
- for 语句可以转换成 while 结构

```
expr1;  
while(expr2)  
{  
    循环体语句 ;  
    expr3;  
}
```

例子

```
例：#include<stdio.h>
main( )
{   int i=0;
    for(i=0;i<10;i++)
        putchar(`a'+i);
}
```

```
例：#include<stdio.h>
main( )
{   int i=0;
    for(;;i<10;i++)
        putchar(`a'+i);
}
```

运行结果： abcdefghij

```
例：#include<stdio.h>
main( )
{   int i=0;
    for(;;i<10;putchar(`a'+i),i++)
        ;
}
```

练习 1： 将输入的正整数按逆序输出。 P91

练习 2： 从键盘输入若干学生的成绩，当输入负数时结束输入，统计并输出最高成绩和最低成绩。 P92

循环的嵌套

- ❖ 三种循环可互相嵌套，层数不限
- ❖ 外层循环可包含两个以上内循环，但不能相互交叉
- ❖ 嵌套循环的执行流程

❖ 嵌套循环的跳转

禁止：

- 从外层跳入内层
- 跳入同层的另一循环
- 向上跳转

```
(1) while()
{
    .....
    while()
    {
        .....
    }
    .....
}
```

```
(2) do
{
    .....
    do
    {
        .....
    }while();
    .....
}while();
```

```
(3) while()
{
    .....
    do
    {
        .....
    }while();
    .....
}
```

```
(4) for(;;)
{
    .....
    do
    {
        .....
    }while();
    .....
    while()
    {
        .....
    }
}
```

外循环

内循环

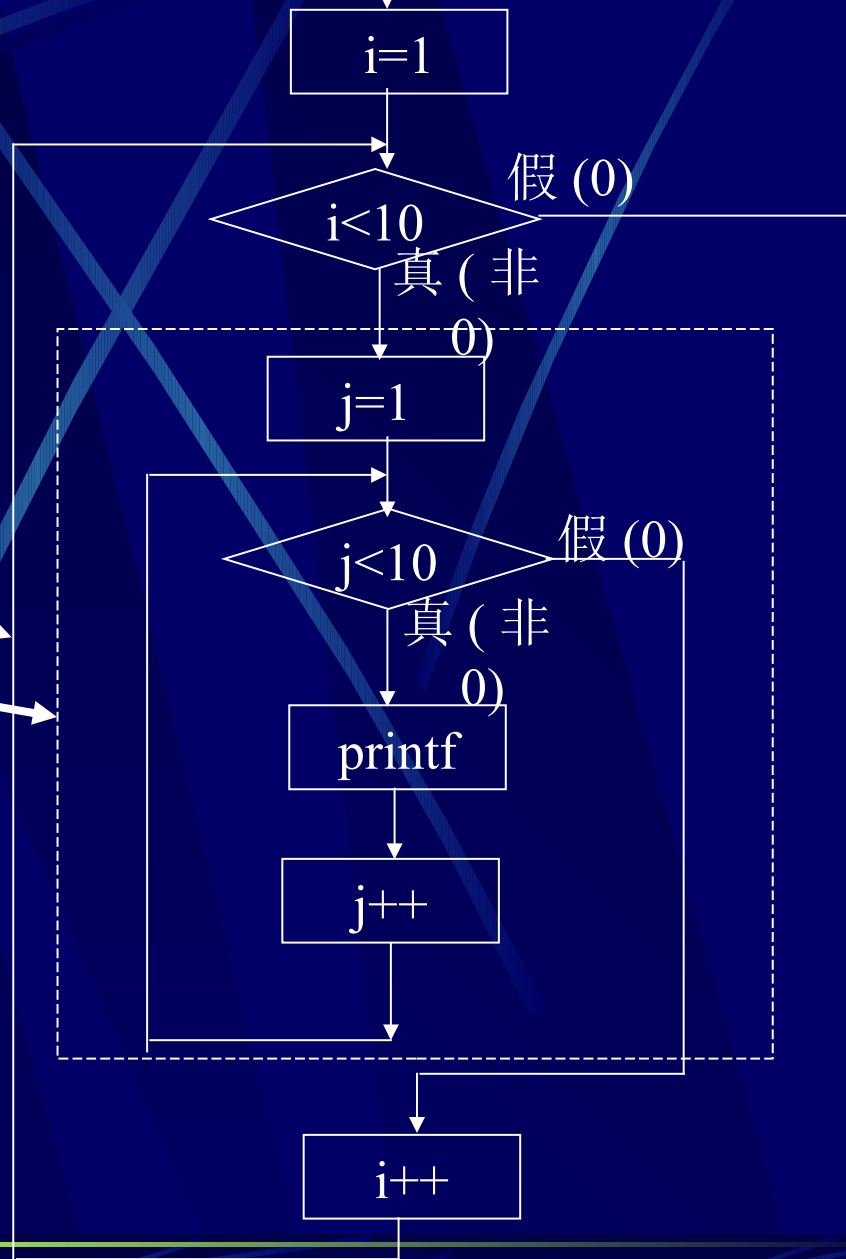
内循环

例 3 输出九九乘法表 P89

```
for(i=1;i<10;i++)  
for(j=1;j<10;j++)  
printf((j==9)?"%4d\n":"%4d",i*j);
```

循环的嵌套 (图解)

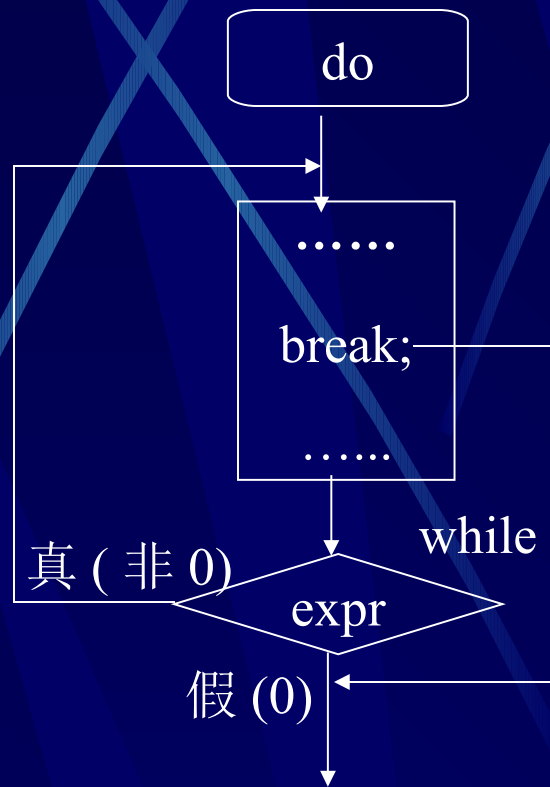
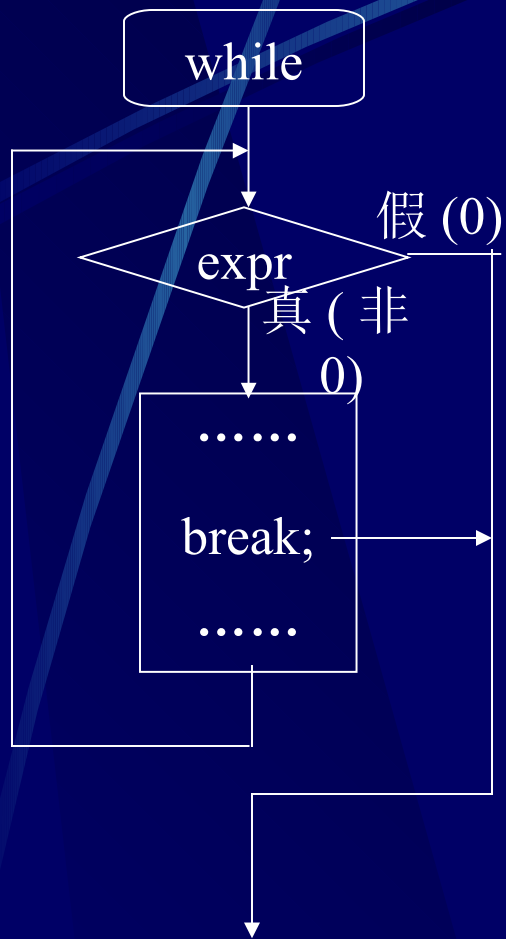
外循环
内循环



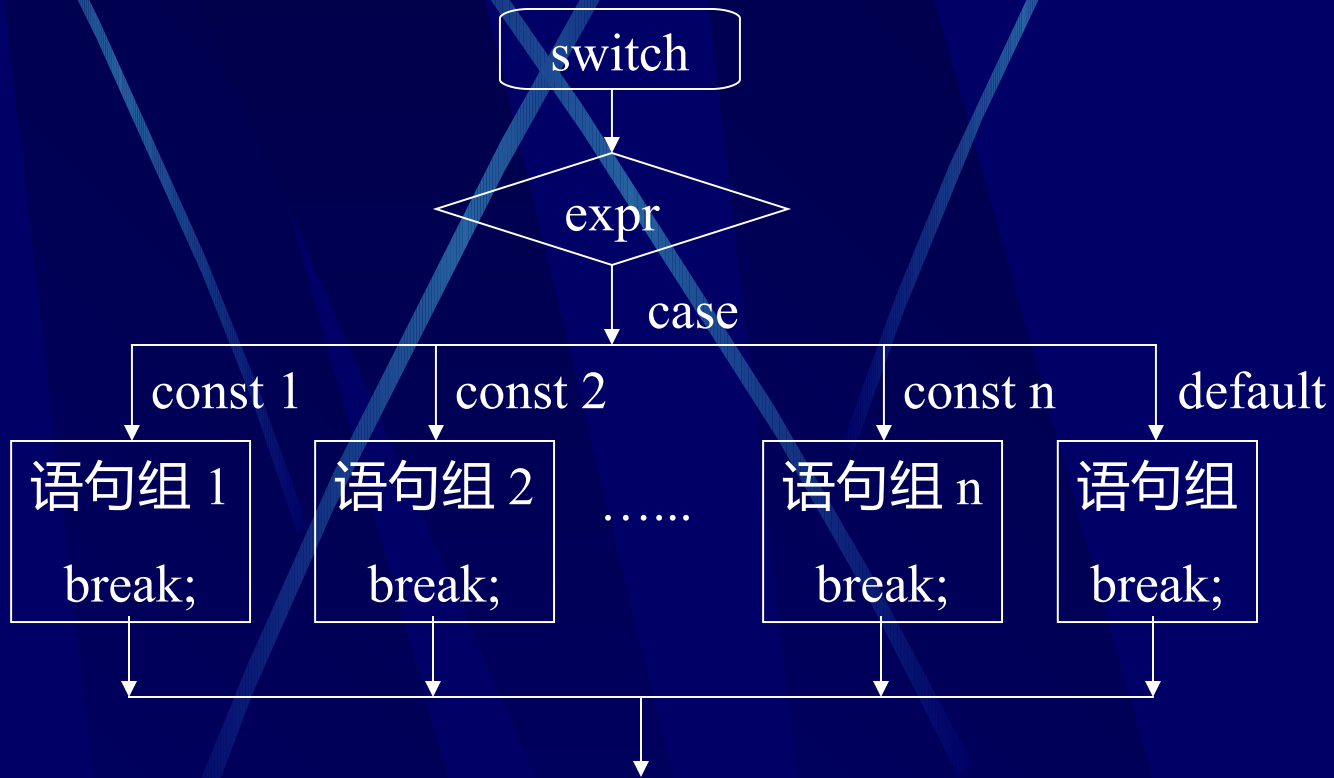
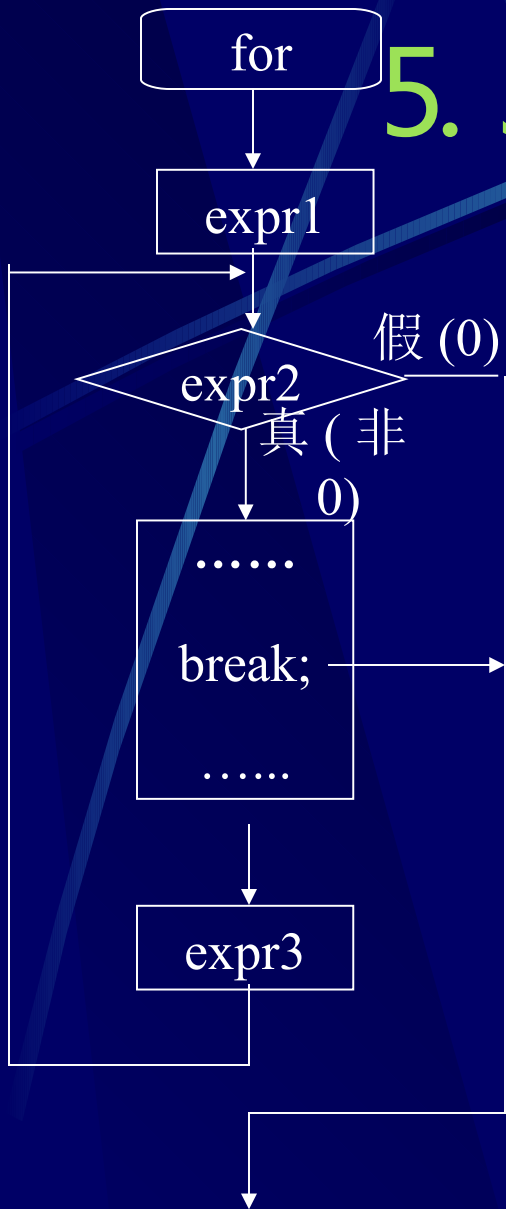
5.3 辅助控制语句

- 例 4：计算 $1+2+3\dots$ 的值，当和 >1000 时结束，并输出和。
 - **break** 语句
 - 功能：在循环语句和 **switch** 语句中，终止并跳出循环体或开关体
 - 说明：
 - **break** 只能终止并跳出最近一层的结构
 - **break** 不能用于循环语句和 **switch** 语句之外的任何其它语句之中

5.3 辅助控制语句（图解 1）



5.3 辅助控制语句（图解2）

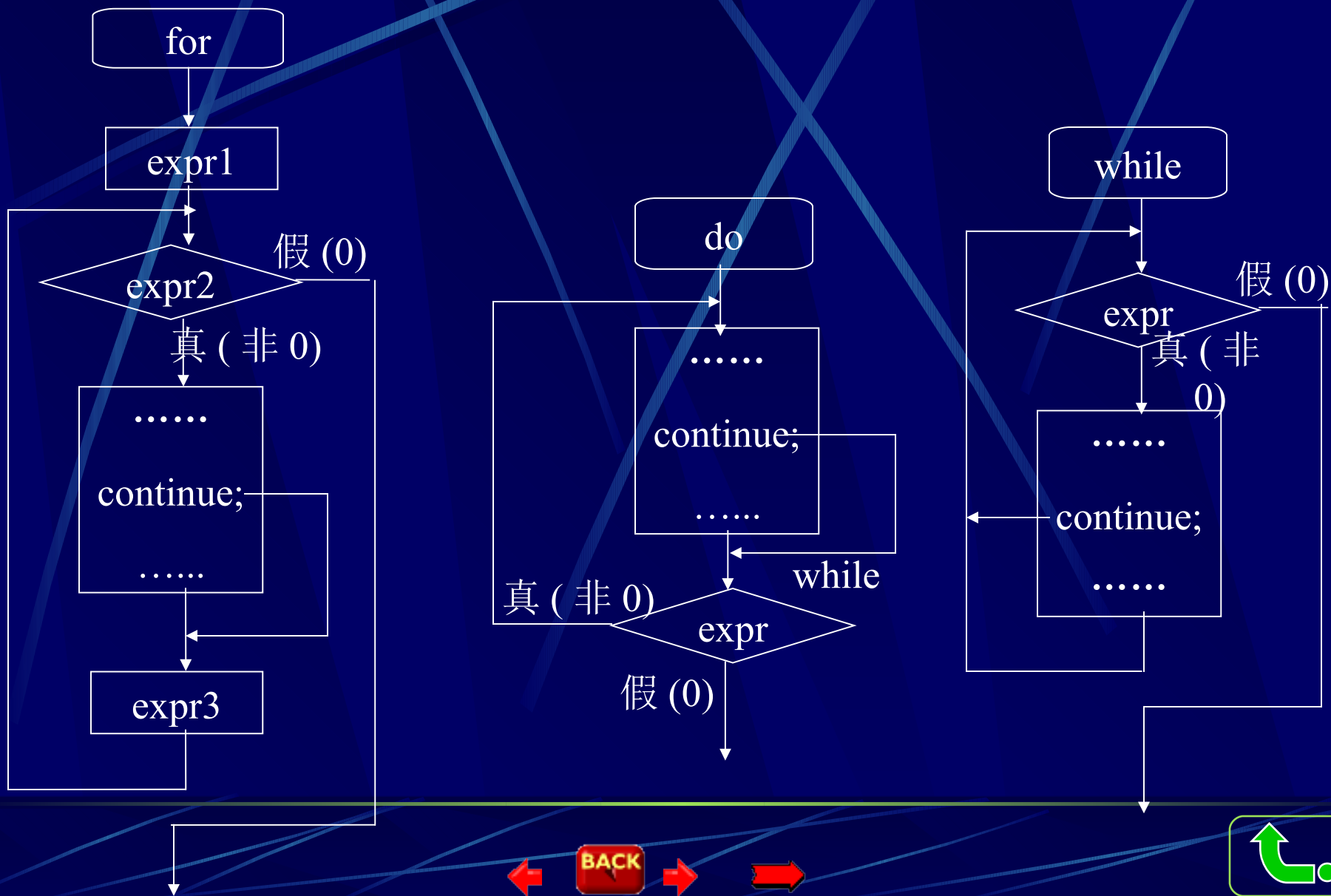


5.3 辅助控制语句

- 例 5：输出 20 以内，不是 5 的倍数的值。

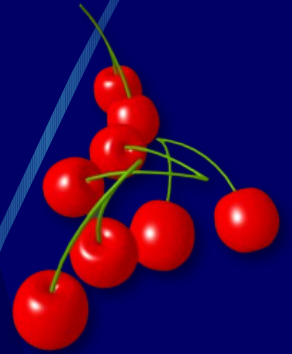
continue 语句

- ❖ 功能：结束本次循环，跳过循环体中尚未执行的语句，进行下一次是否执行循环体的判断
- ❖ 仅用于循环语句中





5.4 回到工作场景



- 通过对以上各小节内容的学习，我们了解了 while 语句、do-while 语句以及 for 语句实现循环的过程之后，现在是时候回到我们 5.1 节的工作场景任务中了。
- 运用循环结构控制的结束条件应该是本金和利息之和超过 150 万；本金和利息之和通过变量 sum 得到，即 $sum += sum * 0.07$ ，循环最大次数可由银行还款最大期限决定，即通过 i 来控制最大循环次数。此处先用 for 循环实现该场景。

● 程序代码如下：

```
● #include <stdio.h>
● main()
● {
● int i;
● float sum = 1000000;           // 初始化为原贷款金额
● for(i=1; i<=30; i++)
● {
● sum += sum*0.07;             // 累加得到最终还款额
● if(sum > 1500000)           // 程序出口，本息之和大于150万
● 时，
●                               // 不满足题意
● break;
● }
● printf("最大年限为: %d\n", i-1); // 注意为i-1，而非i
● }
```

● 程序运行结果如右图所示。



5.7 工作实训营

● 1. 训练内容

- 编写程序，问题描述：某公司为了防止数据信息泄露，采取了数据加密，数据是小于8位的整数。加密规则如下：将每位数字都加上5，再用和除以10的余数代替该数字，最后将第一位和最后一位数字交换。如输入1234，则应输出9786。请你帮助实现加密。

● 2. 训练目的

- 掌握各类循环结构的使用方法以及各自的适用范围。

本章小结

本章介绍了循环结构设计。在C语言中，用 `while` 语句、`do-while` 语句和 `for` 语句均能实现循环控制。结合使用 `break` 语句、`continue` 语句，还可以改变程序的执行流程，提前退出循环或提前结束本次循环。

本章的知识结构如下图所示。

