

## 1.2.3 循环语句

### ▶ 课前自主预习

## ▶ 学习目标

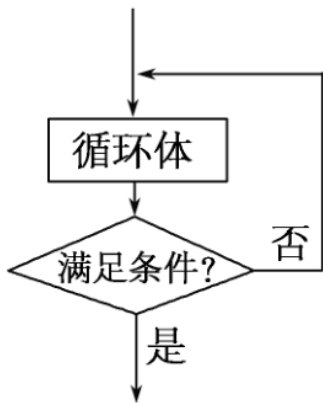
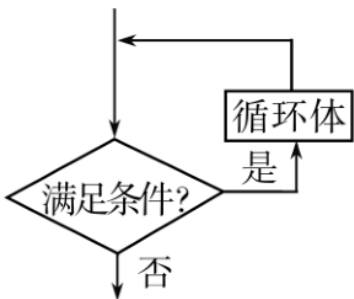
1. 理解循环语句的概念，并驾驭其结构.
2. 会应用条件语句和循环语句编写程序.
3. 通过对现实生活情境的探究，体会应用计算机解决数学问题的便利简捷.

## ▶ 要点梳理

### 1. 循环语句

循环语句与程序框图中的循环结构相对应，一般程序设计语言中都有直到型(UNTIL)和当型(WHILE)两种循环语句结构，分别对应于程序框图中的直到型和当型循环结构.

### 2. 两种循环语句的对比

名称	直到型	当型
程序结构框图		
格式	DO 循环体 LOOP UNTIL 条件	WHILE 条件 循环体 WEND

<p>执行步骤</p>	<p>先执行一次 DO 和 UNTIL 之间的循环体,再对 UNTIL 后的条件进行判断,如果条件不符合,继续执行循环体,然后再检查上述条件,如果条件仍不符合,再次执行循环体,直到某一次条件符合为止.这时不再执行循环体,跳出循环体执行 UNTIL 后面的语句</p>	<p>先判断条件的真假,如果条件符合,就执行 WHILE 和 WEND 之间的循环体,然后再检查上述条件,如果条件仍符合,再次执行循环体,这个过程反复进行,直到某一次条件不符合为止,这时不再执行循环体,跳出循环体,执行 WEND 后面的语句</p>
-------------	---	--

## 思考诊断

1. 循环语句与条件语句有何关系?

**[提示]** 循环语句中肯定有条件语句,条件语句是循环语句的一部分,离开条件语句,循环语句无法循环.但条件语句可以脱离循环语句单独存在,可以不依靠循环语句独立地解决问题.

2. 编写程序时,什么状况下运用循环语句?

**[提示]** 在问题处理中,对不同的运算对象进行若干次相同运算或处理时,一般用到循环结构,在编写程序时要用到循环语句.

3. 推断正误.(正确的打“√”,错误的打“×”)

(1) UNTIL 语句中先进行条件推断,再执行循环体; WHILE 语句中先执行循环体,再进行条件推断. ( )

(2) 循环语句中肯定有条件语句,条件语句中肯定有循环语句. ( )

(3) 直到型循环语句和当型循环语句执行循环体的次数都可能是零. ( )

**[提示]** (1) × UNTIL 语句中先执行循环体,再进行条件推断; WHILE 语句中先进行条

件推断，再执行循环体.

(2)× 循环语句中肯定有条件语句，但条件语句中不肯定有循环语句.

(3)× 直到型循环语句中执行循环体的次数大于或等于 1.

## ▶ 课堂互动探究

### 题型一 UNTIL 语句及应用

**【典例 1】** (1)用 UNTIL 语句写出求  $1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \cdots + \frac{1}{999} - \frac{1}{1000}$  的程序.

(2)依据下列程序，画出相应的程序框图.

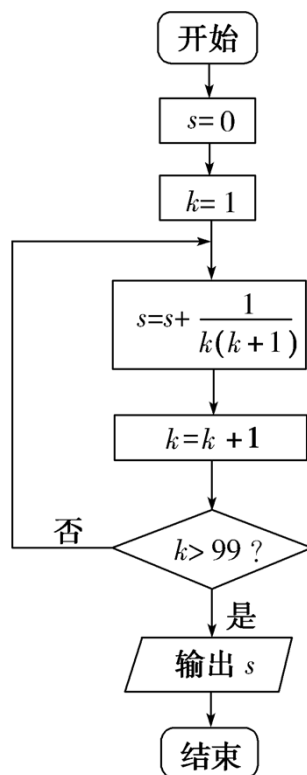
```
s=0
k=1
DO
s=s+1/(k*(k+1))
k=k+1
LOOP UNTIL k > 99
PRINT s
END
```

[思路导引] (1)这是一个累加求和问题，共有 1000 项相加，可设计一个记数变量，一个累加变量，用循环结构实现这一算法；(2)依据语句转化为程序框图.

**[解]** (1)程序如下：

```
s=0
i=1
DO
s=s+((-1)^(i-1))/i
i=i+1
LOOP UNTIL i > 1000
PRINT s
END
```

(2)程序框图如图所示.



## 名师提醒

对 UNTIL 语句的几点说明

(1) 直到型循环语句中先执行一次循环体，再推断条件是否满意，以确定接着循环还是退出循环。

(2) 循环次数的限制往往是推断条件，在循环体内要限制条件的变更，否则会陷入“死循环”。

(3) 限制循环次数的变量要综合考虑初始化时和 LOOP UNTIL 后两处，若初始值为 1，则循环体中累加，若初始值为循环的次数，则循环体中递减。

[针对训练 1] 下面是求满足  $1+3+5+\dots+n>2024$  的最小自然数  $n$  的程序框图，试把它设计成程序。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。

如要下载或阅读全文，请访问：

<https://d.book118.com/075313001222012010>