

### 第3课 行走天涯——条件判断

#### 任务导航

玛丽已经能够自右向左的行走了，但是当他走过窗体左边框后就从我们的视野中消失了，而且玛丽在遇到悬崖时既没有跳跃的动作也没有坠落的效果。这些都是因为我们的程序中没有编写玛丽跳跃和坠落的程序段落所造成的。

本课，我们要实现玛丽从左边框消失后，又从右边框出现的效果，还要编写如图 3.1 所示，能实现玛丽坠落的程序段落。

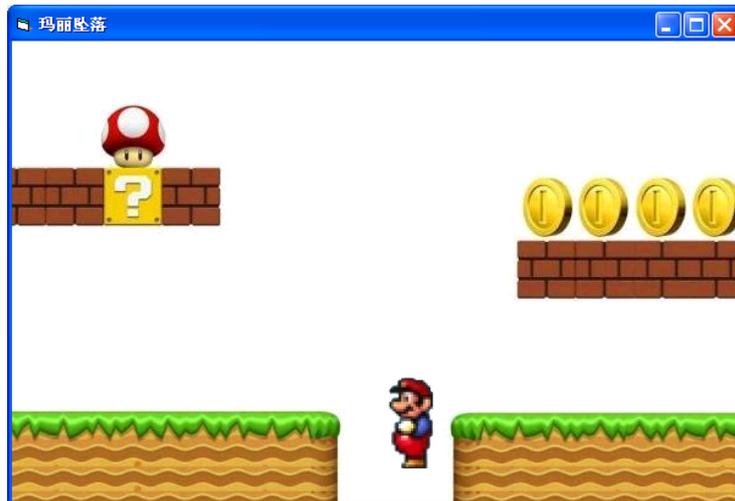


图 3.1 玛丽坠落的程序运行界面

Follow Me

#### 1. 编写实现“玛丽左出右进”的程序

##### 任务分析

智多星

对象的 Width 和 Height 属性

Width 属性值：对象的宽。

Height 属性值：对象的高。

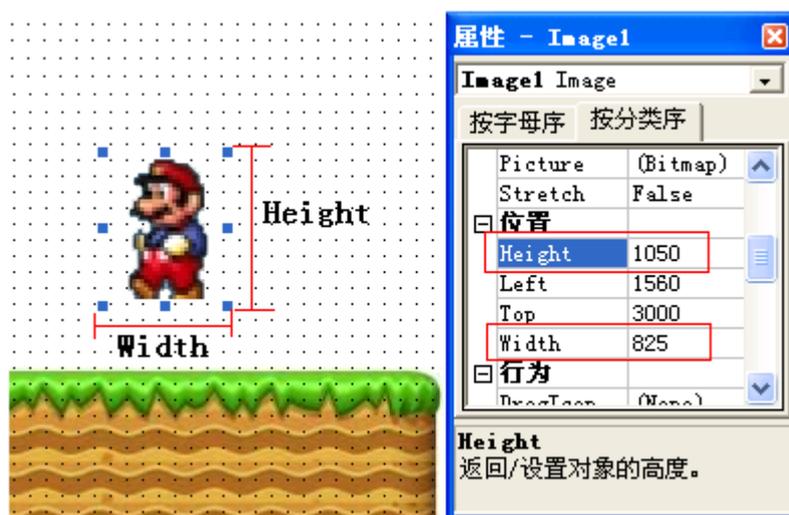


图 3.2 对象的大小

例如：

玛丽图像框的宽为：Image1.Width

玛丽图像框的高为: Image1.Height

- 当玛丽到左边框的距离为玛丽图像宽度的相反数时如图 3.3 所示, 玛丽在窗体上就消失了。

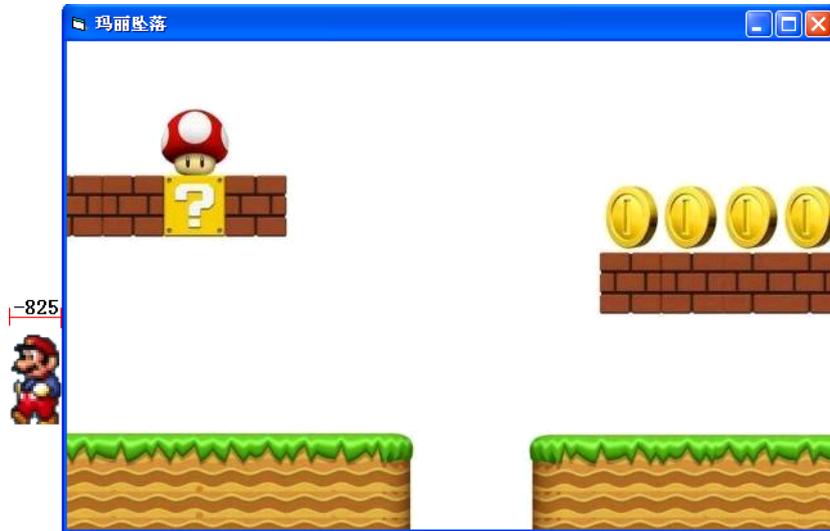


图 3.3 玛丽消失

- 当玛丽从左边框消失时, 让“玛丽到左边框的距离=窗体宽-玛丽图像宽”时, 玛丽就会出现在窗体的右侧, 如图 3.4 所示。



图 3.4 玛丽出现

## (2) 编写事件过程代码

根据分析, 可以简单地将“玛丽左出右进”的过程叙述为:

**如果** 玛丽到左边框的距离小于等于玛丽宽的相反数 **则**

让玛丽到左边框的距离为窗体的宽减去玛丽的宽

**否则**

玛丽左移一段距离

**结束判断**

以上判断过程可以写为以下程序代码:

```

Rem 玛丽左出右进
Private Sub Timer1_Timer()
    If Image1.Left <= -Image1.Width Then ' 玛丽消失的判断
        Image1.Left = Form1.Width - Image1.Width ' 玛丽出现的位置
    Else
        Image1.Left = Image1.Left - 100 ' 玛丽行走
    End If
End Sub

```

(3) 保存、运行并调试程序

将程序以“左出右进.frm”为窗体文件名，“左出右进.vbp”为工程文件名，保存在“第1单元\3 行走天涯\”的文件夹中。

智多星

在玛丽“左出右进”的程序中使用了条件判断的控制语句，条件语句的一般格式为：  
格式 1:

```

If <判断条件> Then
    <程序段 A>
Else
    <程序段 B>
End If
<程序段 C>

```

格式 1 的流程图:

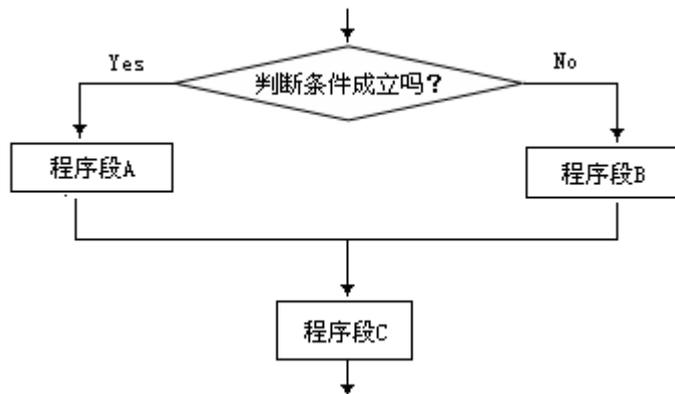


图 3.5 条件语句格式 1 的一般格式和流程图

当程序执行格式 1 的 If 语句时首先判断条件是否成立,如果条件成立则执行 Then 后面的程序段 A, 然后执行 End If 后面的程序段 C; 否则执行 Else 后面的程序段 B, 然后执行 End If 后面的程序段 C, 程序流程图, 如图 3.5 所示。

格式 2:

```

If <判断条件> Then
    <程序段 A>
End If
<程序段 B>

```

格式 2 的流程图:

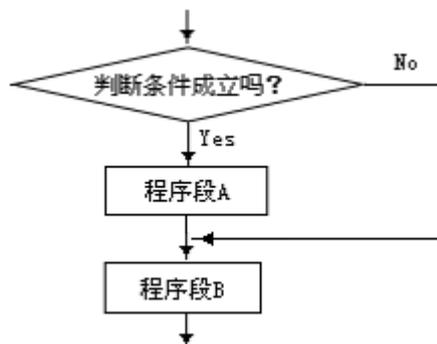


图 3.6 条件语句格式 2 的一般格式和流程图

当程序执行格式 2 的 If 语句时, 如果条件成立则执行 Then 后面的程序段 A, 然后执行

End If 后面的程序段 B；否则越过程序段 A，直接执行 End If 后面的程序段 B，程序流程图，如图 3.6 所示。

条件语句体现了程序的分支结构，程序的分支结构是结构化程序设计三种基本结构之一，也是重要的程序结构。

- 条件语句中的判断条件，是关系表达式或逻辑表达式。
  - 关系式表达式的运算符见表 3-1。

表 3-1 关系表达式运算符表

运算	运算符	运算	运算符
大于	>	大于或等于	>=
小于	<	小于或等于	<=
等于	=	不等于	<>

关系表达式的六种运算为同级运算。关系表达式成立时其值为 True，关系式不成立时其值为 False。

- 逻辑表达式的常用运算符见表 3-2。

表 3-2 逻辑表达式常用运算符表

运算级别	运算	运算符	说 明
高	非	Not	Not 右侧的关系式的值为 True 时表达式的值为 False
中	与	And	And 连结的关系式的值都为 True 时表达式值为 True。
低	或	Or	Or 连结的关系式的值都为 False 时表达式值为 False。

## 2. 编写实现“玛丽坠崖”的程序

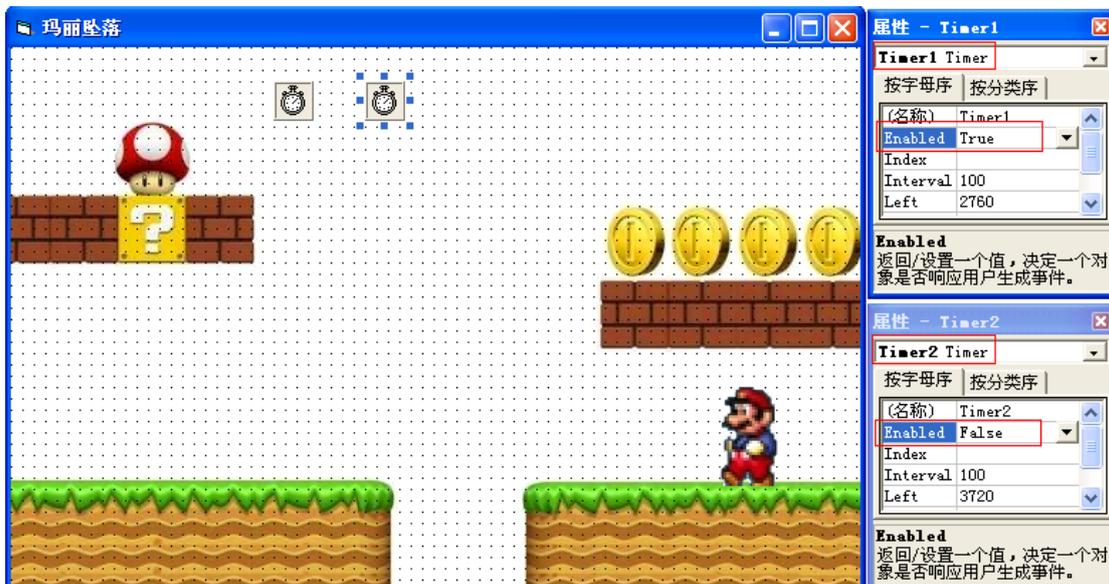
### 任务分析

玛丽坠崖的动作效果，可以看作是玛丽行走和玛丽坠落两个动作的组合。程序首先执行“玛丽行走”的事件过程，当玛丽走到悬崖外边缘时，停止“玛丽行走”程序的执行，转而执行“玛丽坠落”的事件过程。

#### (1) 设计用户界面、设置对象属性

步骤 1：在窗体上添加第二个计时器 Timer2 用于控制玛丽的坠落。

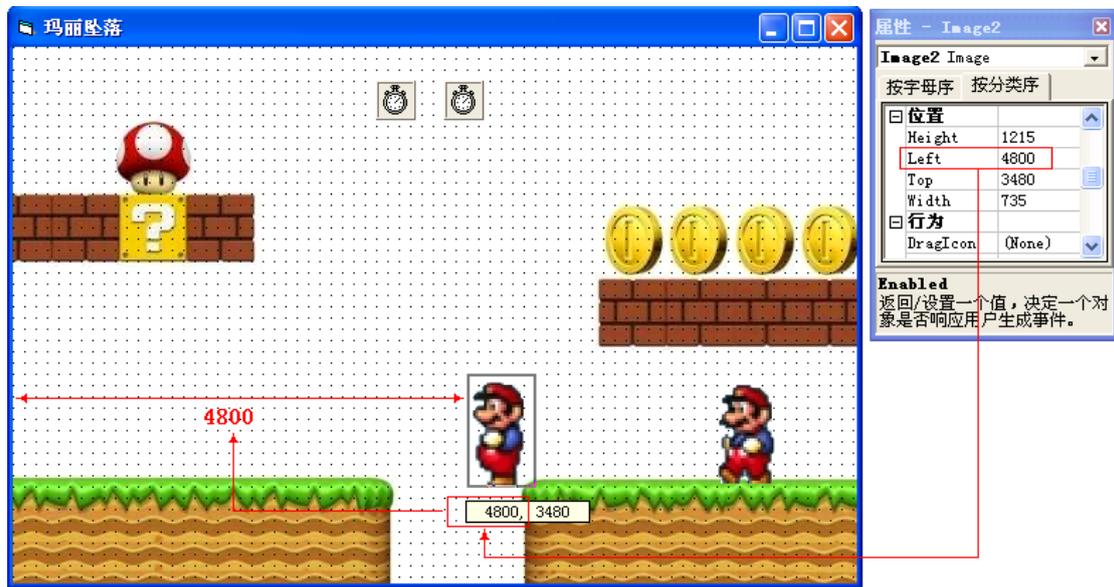
步骤 2：设置计时器 Timer1 的 Interval 属性值为 100，如图 3.7 所示



### 3.7 计时器属性设置

步骤 3: 设置计时器 Timer2 的 Interval 属性值为 100, Enabled 属性值为 False。

步骤 4: 在窗体中添加图像框 Image2, 并装载“前景 1”的玛丽直立图片。



### 3.8 玛丽坠落的条件

将玛丽直立图片 (Image2) 置于悬崖外边缘, 在 Image2 上按下鼠标左键可以看见玛丽直立图片的位置提示框, 如图 3.8 所示, 从属性窗口也可以找到该对象位置的属性值。

步骤 5: 选中“直立玛丽”即 Image2, 在属性窗口将 Visible 的属性值设为 False。  
智多星

Visible 是对象的可见性属性, 当 Visible 的属性值为 True 时对象可见, 当 Visible 的属性值为 False 时对象不可见。

例如:

让玛丽行走图像不可见的代码为 `Image2.Visible=False`

让玛丽直立图像可见的代码为 `Image2.Visible=True`

属性设置如表 3-3

3-3 “玛丽坠崖”属性设置表

对象	属性名	属性值	属性说明
Form1 (窗体)	Caption	玛丽坠崖	显示在窗体标题栏上
	Picture	(Bitmap)	导入“背景 3.gif”
Image1 (图像框 1)	Picture	(Bitmap)	导入“玛丽 2.gif”(行走图片)
Timer1 (计时器 1)	Interval	100	每隔 0.1 秒产生一次 Timer 事件
Image2 (图像框 2)	Picture	(Bitmap)	导入“玛丽 1.gif”(直立图片)
	Visible	False	不可见
Timer2 (计时器 2)	Interval	100	每隔 0.1 秒产生一次 Timer 事件
	Enabled	False	不可用

#### (2) 编写事件过程代码

##### ① 玛丽行走的事件过程

从对图 3.8 所示的分析, 不难看出如果玛丽到左边框的距离大于 4800 时, 则玛丽行走, 否则玛丽坠落。如果我们将玛丽行走和坠落分解为两个事件过程, 则玛丽行走的事件过程应

首先设为有效，此时玛丽坠落的事件过程应设为无效。当玛丽走到悬崖边缘时满足了坠落条件时应将玛丽坠落的事件过程设为有效，而将玛丽行走的事件过程设为无效。

Rem 玛丽行走

```
Private Sub Timer1_Timer()  
    If Image1.Left > 4800 Then ' 坠落判断  
        Image1.Left = Image1.Left - 100 ' 行走  
    Else  
        Timer2.Enabled = True ' 玛丽坠落有效  
        Image2.Visible = True ' 直立玛丽可见  
        Timer1.Enabled = False ' 玛丽行走无效  
        Image1.Visible = False ' 行走玛丽不可见  
    End If  
End Sub
```

## ② 玛丽坠落的事件过程

由以上“玛丽行走”的程序代码中不难看出，当满足“玛丽坠落”条件时，玛丽行走的事件过程被设置为无效，玛丽行走的事件过程被设置为有效。此时玛丽坠落的事件被触发执行，实现玛丽坠落的过程。

DIY

请参考玛丽直行的程序，编写玛丽坠落的程序代码。

Rem 玛丽坠落

```
Private Sub Timer2_Timer()  
    _____ ' 玛丽落下  
End Sub
```

## (3) 保存、运行并调试程序

- 将玛丽行走和玛丽坠落的程序代码作为一个窗体文件，以“玛丽坠崖.frm”为窗体文件名，以“玛丽坠落.vbp”为工程文件名，保存在“第1单元\3 行走天涯\”的文件夹中。
- 在调试过程中遇到以下错误你会如何调试？
  - 程序运行后玛丽不动
  - 程序运行后玛丽向左下方移动
  - 程序运行后玛丽坠落的方向相反
  - 程序运行后出现了两个玛丽图像

知识着陆：

### 1. 几个常用的属性

- ① 位置属性：Left 的值表示对象到容器左边框距离，Top 的值表示对象到容器上边框的距离。
- ② 大小属性：Width 的值表示对象的宽，Height 的值表示对象的高。
- ③ 可用性属性：Enabled 的值为 True 时对象可用，Enabled 的值为 False 时对象不可用。
- ④ 可见性属性：Visible 的值为 True 时对象可见，Visible 的值为 False 时对象不可见。

### 2. 静态属性设置与动态属性设置

在属性窗口设置属性称为静态设置属性，在程序代码中设置属性称为动态设置属性，代码中设置的属性是在程序运行过程中设置的，代码书写的一般格式为：

对象名. 属性名 = 属性值

### 3. 条件语句

根据条件成立与否决定程序执行的内容的控制语句是条件语句，也称分支语句，条件语句的一般格式为：

格式 1: If <判断条件> Then <程序段 A> Else <程序段 B> End If <程序段 C>	格式 2: If <判断条件> Then <程序段 A> End If <程序段 B>
--	---

视野拓展：

编程解决问题的一般过程

编程解决问题的一般过程是：分析问题，算法设计，编写程序，调试程序。

#### (1) 分析问题

分析所需的已知条件信息。需要将已知的信息输入给计算机。

分析如何解决问题。计算机要如何处理输入的原始信息，如何进行判断，最终输出结果。

#### (2) 算法设计

算法 (Algorithm) 是解决问题的方法和步骤。算法是程序设计的“灵魂”。算法的描述可以使用自然语言直接叙述，也可采用伪代码或者流程图等描述。

#### (3) 编写程序

算法只是一个解决问题步骤的集合。如果要用计算机按照算法真正的去执行，就需要用程序设计语言按算法描述编写程序，计算机执行程序从而完成任务。

#### (4) 调试程序

经过反复的调试、检验、去除程序中存在的错误并完善程序，直到运行结果正确。

创作天地：

1. 修改“左出右进”的程序，使玛丽按“右出左进”的方式移动。
2. 修改程序，使小飞机按“上出下进”的方式移动。
3. 修改程序，使小飞机在窗体边宽内左右移动（移到左边框时向右边框移动，移到右边框时向左边框移动）。
4. 根据图 3.9 所示，编写“玛丽越崖”的程序。

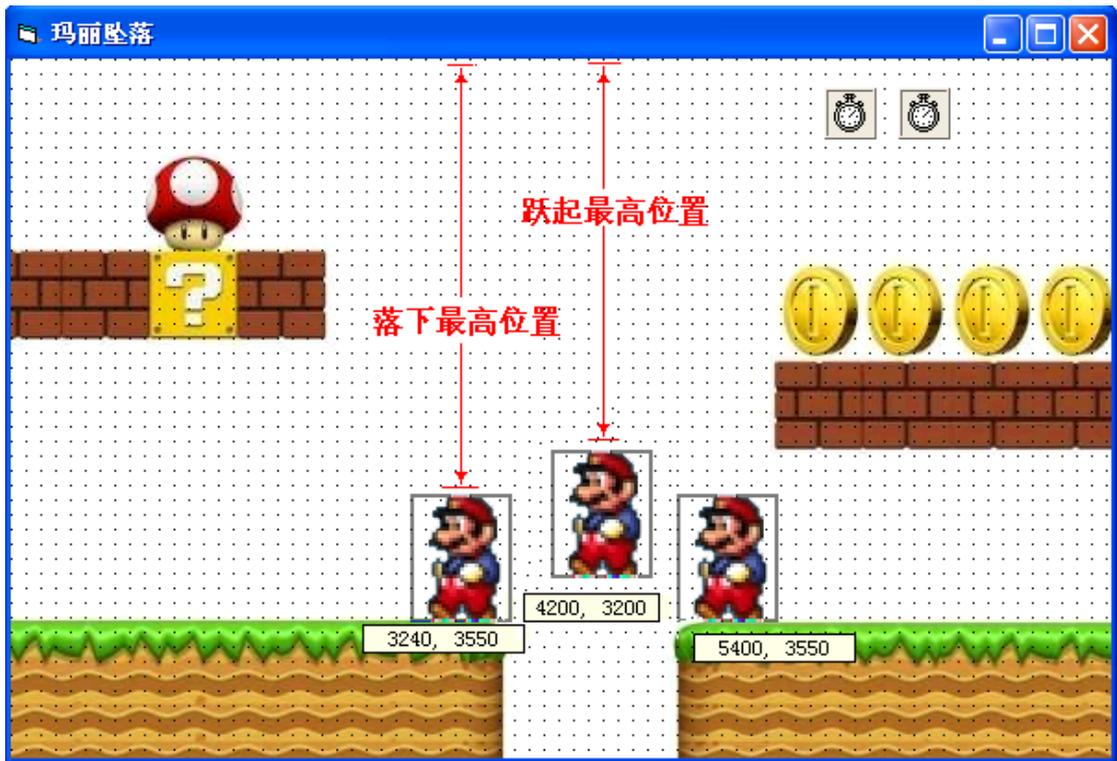


图 3.9 玛丽越崖分析图

提示:添加 Timer3 和 Timer4 计时器分别用于玛丽跃起和玛丽落下的过程控制,参照“玛丽坠崖”的程序,分别设计好跃起和落下的事件过程后,将跃起和落下组合成为“玛丽越崖”的程序。