

第 10 章

動きのあるゲーム

ここでは、オブジェクトの移動、キーボードからの入力判定について学んだ後、動きのある本格的なゲームを作成しましょう。

10.1 オブジェクトの位置

オブジェクトの位置もプロパティです。Location プロパティを見ることでオブジェクトの座標がわかります。また、Location プロパティを変更することでオブジェクトの位置を変更できます。Location プロパティは X 座標と Y 座標がくっついた Point という名前の構造体です。Location.X が X 座標の値、Location.Y が Y 座標の値です。なお、Location プロパティが表している座標は、そのオブジェクトの左上の点の座標です。

【例 1】 PictureBox1 の X 座標と Y 座標の値をそれぞれ TextBox1 と TextBox2 に表示する場合、次のように書きます。

```
TextBox1.Text = PictureBox1.Location.X  
TextBox2.Text = PictureBox1.Location.Y
```

【例 2】 PictureBox1 の座標を (100, 200) に設定する場合、次のように書きます。

```
Dim P As Point  
P.X = 100  
P.Y = 200  
PictureBox1.Location = P
```

【例 3】 PictureBox1 の座標を (100, 200) に設定する場合、次のように書くこともできます。

```
PictureBox1.Location = New Point(100, 200)
```

Location プロパティを用いずに、Left プロパティと Top プロパティによりオブジェクトの X 座標と Y 座標を別々に参照または設定することもできます。なお、Left と Top はプロパティウィンドウに表示されていませんが、Left に値を設定すると Location.X の値が、Top に値を設定すると Location.Y の値が変わります。

【例 4】 PictureBox1 の X 座標と Y 座標の値を、Left と Top プロパティにより、それぞれ TextBox1 と TextBox2 に表示する。

```
TextBox1.Text = PictureBox1.Left  
TextBox2.Text = PictureBox1.Top
```

【例 5】 PictureBox1 の座標を(100, 200)に設定する場合、次のように書くこともできます。

```
PictureBox1.Left = 100
```

```
PictureBox1.Top = 200
```

【例 6】 PictureBox1 を下に 1 ピクセル移動したい場合、次のように書きます。

```
PictureBox1.Top = PictureBox1.Top + 1
```

【例 7】 PictureBox1 を上に 1 ピクセル移動したい場合、次のように書きます。

```
PictureBox1.Top = PictureBox1.Top - 1
```

【例 8】 PictureBox1 を右に 1 ピクセル移動したい場合、次のように書きます。

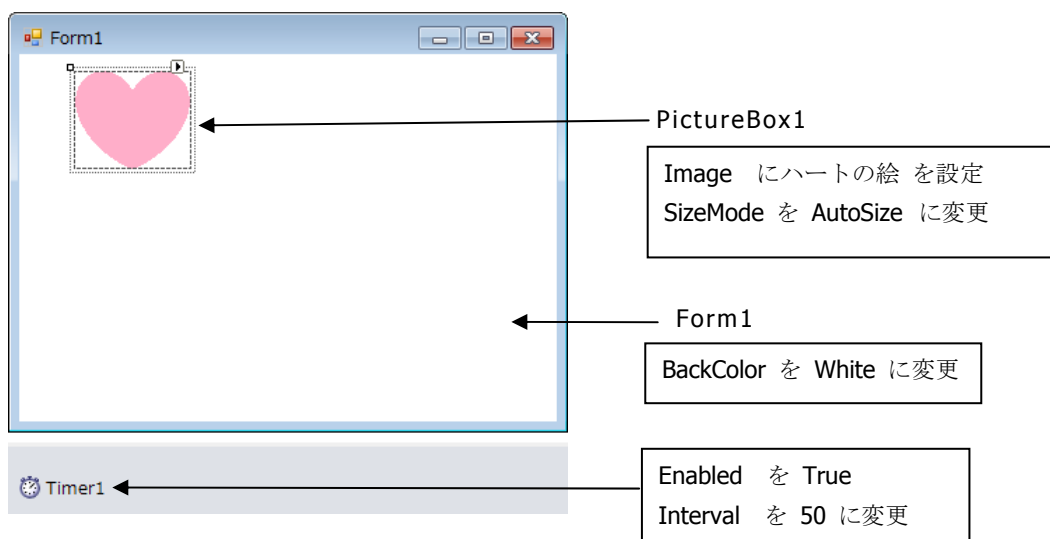
```
PictureBox1.left = PictureBox1.left + 1
```

【例 9】 PictureBox1 を左に 1 ピクセル移動したい場合、次のように書きます。

```
PictureBox1.left = PictureBox1.left - 1
```

Timer を使用して、指定した時間間隔でオブジェクトの位置を更新することでオブジェクトに動きを持たせることができます。

例題 10.1 ハートの絵を描いてピクチャーボックスに絵を貼り付けハートが上から下に向かって動くように見えるプログラムを作りなさい。



```
Private Sub Timer1_Tick(...省略...) Handles Timer1.Tick
```

```
    PictureBox1.Top = PictureBox1.Top + 5
```

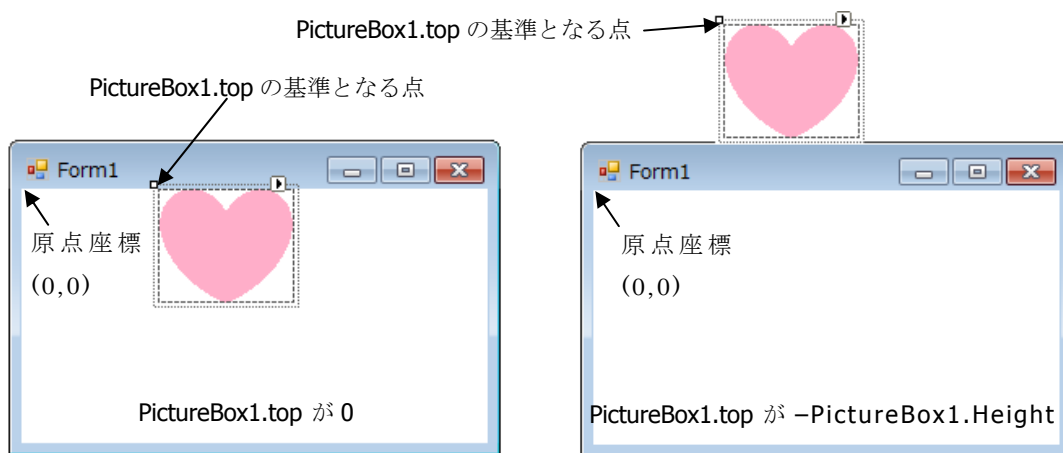
```
    ' PictureBox1を下に5ピクセル移動
```

```
End Sub
```

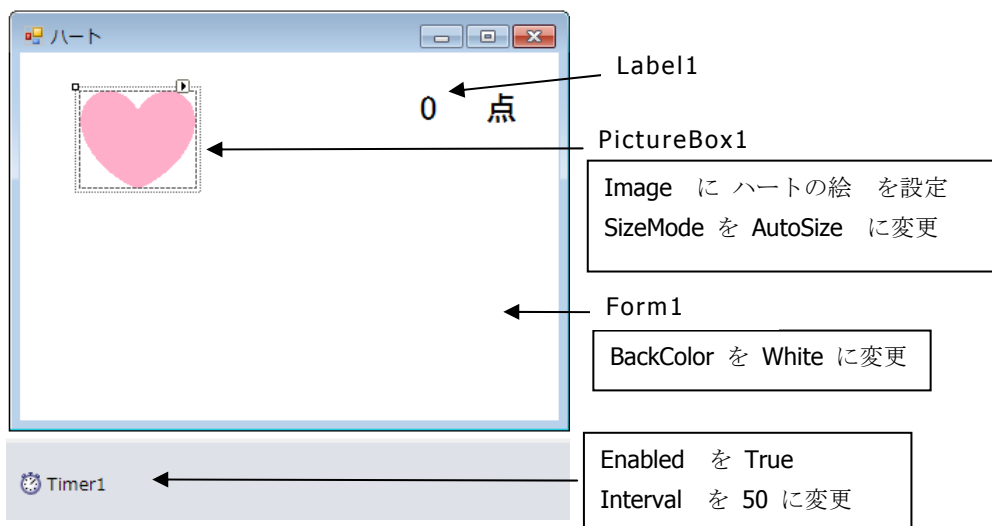
例題 10. 2 ハートの絵が下に移動して見えなくなったらもう一度上から現れるように例題 10. 1のプログラムを改良しなさい。

```
Private Sub Timer1_Tick(...省略...) Handles Timer1.Tick
    If PictureBox1.Top < Me.Height Then      ' PictureBox1のTopの値がForm1の高さ未満の場合
        PictureBox1.Top = PictureBox1.Top + 5    ' PictureBox1を下に5ピクセル移動
    Else                                     ' 条件が成立しなかった場合
        PictureBox1.Top = -PictureBox1.Height    ' PictureBox1を上移動
    End If
End Sub
```

PictureBox1.Height は、PictureBox1 の高さ（Y方向の長さ）を表します。このプログラムでは、PictureBox1 を上に移動させる時に、オブジェクトが完全に見えない位置から表示させるために Top プロパティに `-PictureBox1.Height` を代入しています。



例題 10.3 ハートをクリックするゲームを作りました。



```

Dim a As Integer                                ' 横方向の移動量を記憶する変数
Private Sub Timer1_Tick(...省略...) Handles Timer1.Tick
    If PictureBox1.Top < Me.Height Then          ' PictureBox1のTopの値がForm1の高さ未満の場合
        PictureBox1.Top = PictureBox1.Top + 5    ' PictureBox1を下に5ピクセル移動
        PictureBox1.Left = PictureBox1.Left + a  ' PictureBox1を横方向に移動
    Else                                          ' 条件が成立しなかった場合
        PictureBox1.Top = -PictureBox1.Height    ' PictureBox1を上に移動
        PictureBox1.Left = Rnd() * (Me.Width - PictureBox1.Width) ' PictureBox1の横の位置を決定
        a = 5 * Rnd() - 2                        ' 横方向の移動量を乱数により決定
    End If
End Sub

Private Sub PictureBox1_Click(...省略...) Handles PictureBox1.Click ' PictureBox1をクリックした
    Label1.Text = Label1.Text + 1                ' 点数を1点増やす
    PictureBox1.Top = Me.Height                  ' PictureBox1を下に移動
End Sub

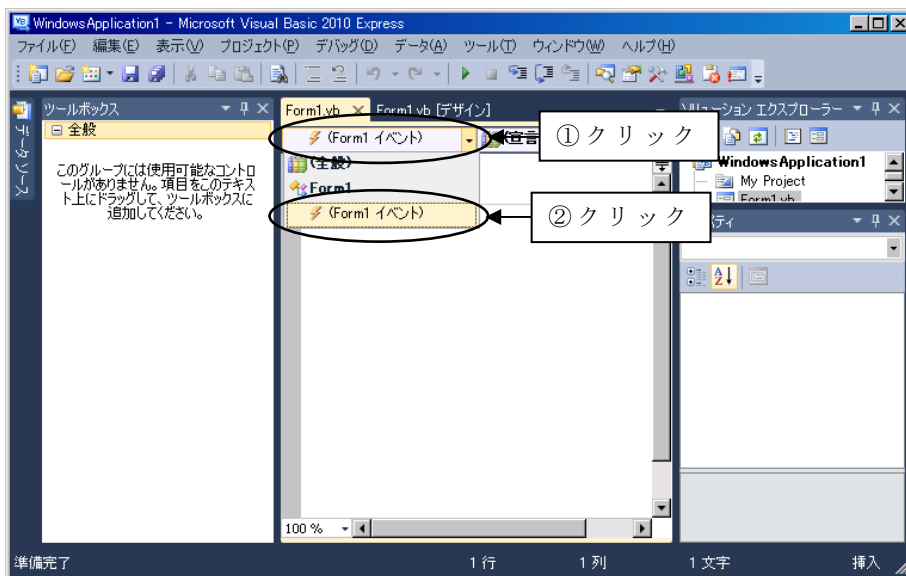
Private Sub Form1_Load(...省略...) Handles MyBase.Load
    Randomize()                                  ' 乱数系列の初期化
    a = 5 * Rnd() - 2                            ' 横方向の移動量を乱数により決定
End Sub

```

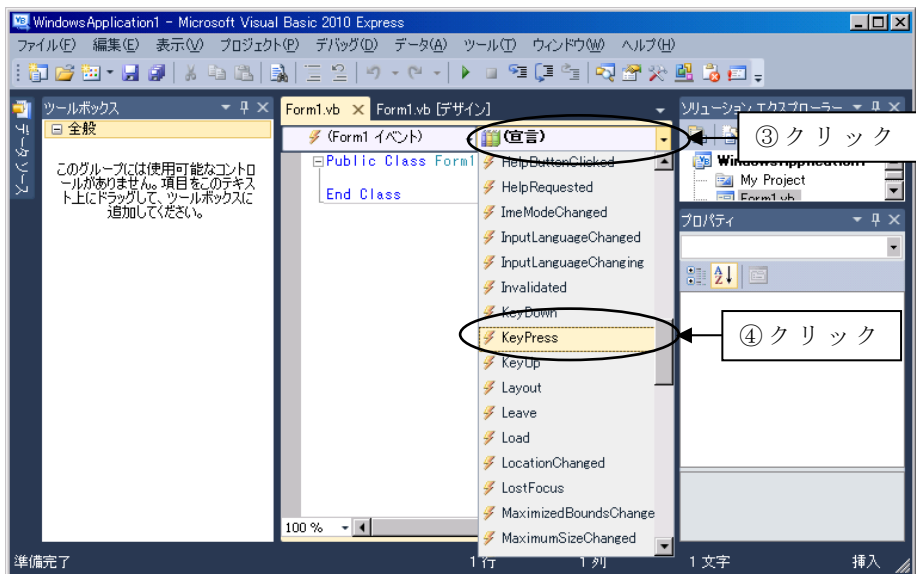
9.2 キーボード入力の判定

キーボードが押されたときに処理を行いたい場合には、**KeyPress**イベントまたは**KeyDown**イベントを用います。**KeyPress**イベントは、文字キーを押したときに発生し、**KeyDown**イベントは文字キーだけでなく矢印キーやファンクションキーを押したときにも発生します。

KeyPressイベントプロシージャを作成する場合、まず、コードウィンドウの上部にある左のプルダウンメニューで**Form1**イベントを選択します。



次に、右のプルダウンメニューから**KeyPress**を選択します。



次のようなコードが自動的に生成されますので、Private Sub Form1_KeyPressとEnd Sub の間に文字キーが押されたときに実行させたいイベントプロシージャを記述します。

```
Private Sub Form1_KeyPress(...省略...) Handles Me.KeyPress
```

```
End Sub
```

KeyPressイベントにおいてどの文字キーが押されたかは、e.KeyCharにより知ることができます。押した文字キーがそのままe.KeyCharの値として入っています。

【例1】文字キーが押されたとき、Label1にその文字を表示させるには、KeyPressのイベントプロシージャに次のプログラムコードを書きます。

```
Private Sub Form1_KeyPress(...省略...) Handles Me.KeyPress
```

```
Label1.Text = e.KeyChar
```

```
End Sub
```

【例2】例1のKeyPressのイベントプロシージャが書かれているときに、キーボードから a のキーを押すとLabel1に a と表示されます。

【例3】例1のKeyPressのイベントプロシージャが書かれているときに、シフトキーを押した状態でキーボードから a のキーを押すとLabel1に A と表示されます。

文字キー以外のキーが押されたときに処理を行いたい場合は、KeyDownイベントを用います。KeyDownイベントにおいてどのキーが押されたかは、e.KeyCodeにより知ることができます。代表的なキーコードを表10.1に示します。

表10.1 代表的なキーコード

アルファベット		数字		テンキー 数字	テンキー 記号	記号	制御キー
A 65	O 79	0 48	0 96	* 106	:* 186	BackSpace	8
B 66	P 80	1 49	1 97	+ 107	;+ 187	Enter	13
C 67	Q 81	2 50	2 98		,< 188	Shift	16
D 68	R 82	3 51	3 99	- 109	-= 189	Ctrl	17
E 69	S 83	4 52	4 100	. 110	.> 190	Alt	18
F 70	T 84	5 53	5 101	/ 111	/? 191	スペース	32
G 71	U 85	6 54	6 102		@` 192	PageUp	33
H 72	V 86	7 55	7 103		[{ 219	PageDown	34
I 73	W 87	8 56	8 104		¥ 220	←	37
J 74	X 88	9 57	9 105]} 221	↑	38
K 75	Y 89				^~ 222	→	39
L 76	Z 90				¥_ 226	↓	40

【例 4】キーが押されたとき、Label1に押された文字を表示させるには、KeyDownのイベントプロシージャに次のプログラムコードを書きます。

```
Private Sub Form1_KeyDown(...省略...) Handles Me.KeyPress
    Label1.Text = e.KeyCode
End Sub
```

【例 5】例 4 のKeyDownのイベントプロシージャが書かれているときに、キーボードの a のキーを押すとLabel1に 65 と表示されます。

【例 6】例 4 のKeyDownのイベントプロシージャが書かれているときに、キーボードのシフトキーを押すとLabel1に 16 と表示されます。

【例 7】例 4 のKeyDownのイベントプロシージャが書かれているときに、キーボードの左矢印キーを押すとLabel1に 37 と表示されます。

【例 8】例 4 のKeyDownのイベントプロシージャが書かれているときに、キーボードの右矢印キーを押すとLabel1に 39 と表示されます。

【例 9】上下左右の矢印キーを押すと、PictureBox1が押した矢印の方向に 1 ピクセル移動するプログラムは次のようになります。

```
Private Sub Form1_KeyDown(...省略...) Handles Me.KeyPress
    If e.KeyCode = 37 Then PictureBox1.Left =PictureBox1.left - 1 ' 左矢印キーに対する処理
    If e.KeyCode = 39 Then PictureBox1.Left =PictureBox1.left + 1 ' 右矢印キーに対する処理
    If e.KeyCode = 38 Then PictureBox1.top =PictureBox1.top - 1 ' 上矢印キーに対する処理
    If e.KeyCode = 40 Then PictureBox1.top =PictureBox1.top + 1 ' 下矢印キーに対する処理
End Sub
```

プログラムを書く場合に、キーコードは数字なので直感的にわかりにくいいため、キーコードを示す定数が用意されています。代表的なキーコード定数を表10.2に示します。

表10.2 代表的な制御キーを表すコード定数

BackSpace	Keys.Back	Alt	Keys.Menu	←	Keys.Left
Enter	Keys.Enter	スペース	Keys.Space	↑	Keys.Up
Shift	Keys.ShiftKey	PageUp	Keys.PageUp	→	Keys.Right
Ctrl	Keys.ControlKey	PageDown	Keys.PageDown	↓	Keys.Down

【例 1 0】 上下左右の矢印キーが押されるとPictureBox1が押した矢印の方向に1ピクセル移動するプログラムを、キーコード定数を用いて書くと次のようになります。

```
Private Sub Form1_KeyDown(...省略...) Handles Me.KeyPress
```

```
    If e.KeyCode = Keys.Left Then PictureBox1.Left = PictureBox1.Left - 1 ' 左矢印キーに対する処理
```

```
    If e.KeyCode = Keys.Right Then PictureBox1.Left = PictureBox1.Left + 1 ' 右矢印キーに対する処理
```

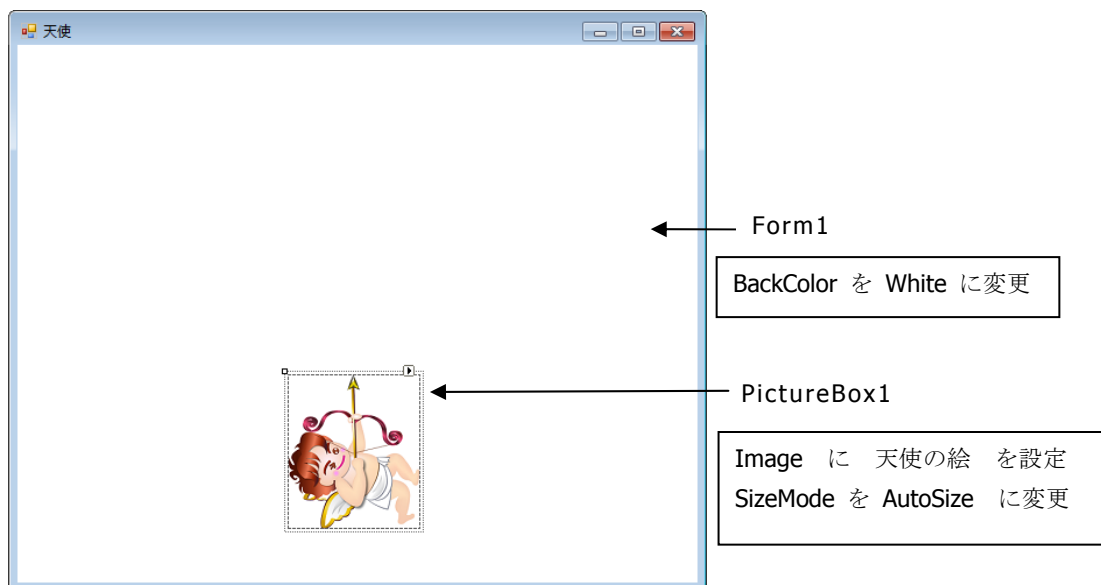
```
    If e.KeyCode = Keys.Up Then PictureBox1.Top = PictureBox1.Top - 1 ' 上矢印キーに対する処理
```

```
    If e.KeyCode = Keys.Down Then PictureBox1.Top = PictureBox1.Top + 1 ' 下矢印キーに対する処理
```

```
End Sub
```

< Point > KeyPress イベントプロシージャ内で押されたキーを知るのに、e.KeyChar を使います。また、KeyDown イベントプロシージャ内では e.KeyCode を使います。

例題 10. 4 天使の絵を描いてピクチャーボックスに絵を貼り付け、矢印キーにより動かしてみましょう。



```
Private Sub Form1_KeyDown(...省略...) Handles Me.KeyDown
```

```
    If e.KeyCode = Keys.Left Then PictureBox1.Left = PictureBox1.Left - 5 ' 左矢印キーに対する処理
```

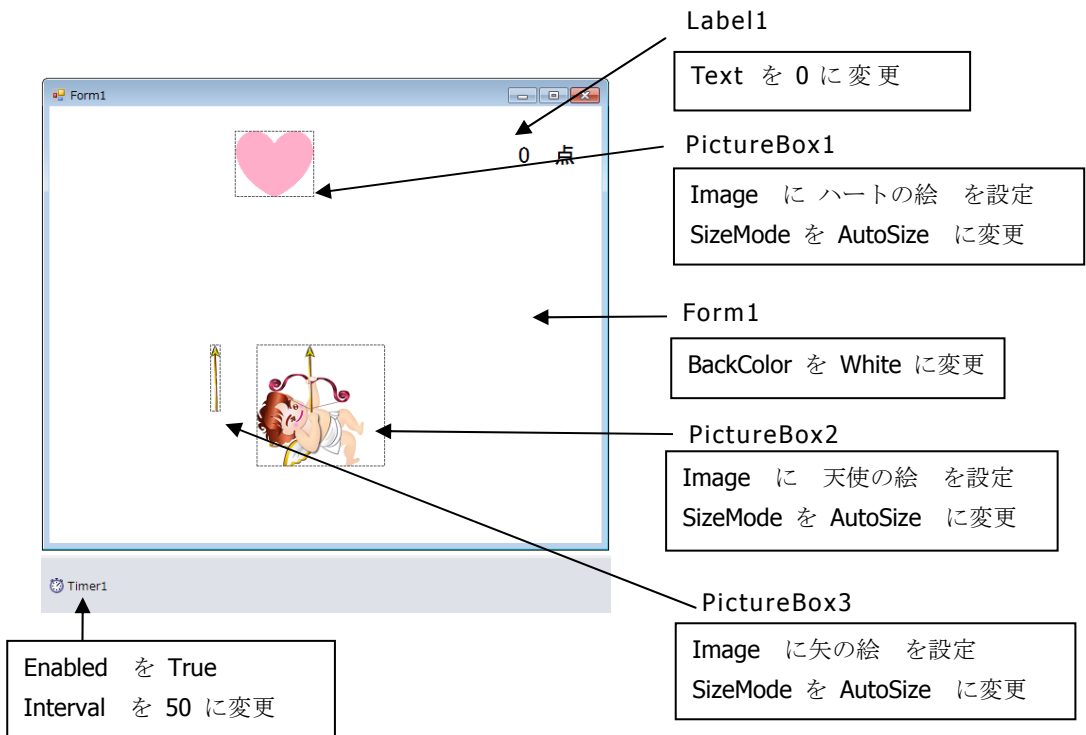
```
    If e.KeyCode = Keys.Right Then PictureBox1.Left = PictureBox1.Left + 5 ' 右矢印キーに対する処理
```

```
    If e.KeyCode = Keys.Up Then PictureBox1.Top = PictureBox1.Top - 5 ' 上矢印キーに対する処理
```

```
    If e.KeyCode = Keys.Down Then PictureBox1.Top = PictureBox1.Top + 5 ' 下矢印キーに対する処理
```

```
End Sub
```


例題 10.5 飛んでくるハートを、天使の矢により射抜くゲームを作りなさい。このゲームは、ハートを射抜くと得点が増えていきます。なお、天使は矢印キーにより操作し、矢はスペースキーで発射するものとします。



プログラムにおいて、ハートと天使の衝突と矢命中の判定部分のif文内で用いた数値は、使用した絵の大きさとどの部分を命中と判断するかを考慮に入れて決めています。なお、今回使用したピクチャーの大きさは以下のとおりです。

表 10.2 使用した絵の大きさ

	横	縦
ハート	84ピクセル	70ピクセル
天使	114ピクセル	133ピクセル
矢	10ピクセル	70ピクセル

Dim a As Integer 'ハートの横方向の移動量を記憶する変数
 Dim m As Integer '矢の状態を記憶する変数

Private Sub Timer1_Tick(...省略...) Handles Timer1.Tick

```

  If PictureBox1.Top < Me.Height Then 'フォーム内にハートがある
    PictureBox1.Top = PictureBox1.Top + 2 'ハートをを下に移動
    PictureBox1.Left = PictureBox1.Left + a 'ハートをを横に移動
  Else 'フォームから外に出たとき
    PictureBox1.Top = -PictureBox1.Height 'ハートを上に移動
    PictureBox1.Left = Rnd() * (Me.Width - PictureBox1.Width) 'ハートの横の位置を決定
    a = 5 * Rnd() - 2
  
```

End If

If m = 1 Then 'もし、矢が発射状態ならば

PictureBox3.Top = PictureBox3.Top - 3 '矢を上へ移動

```

  If PictureBox1.Top + 30 > PictureBox3.Top And PictureBox1.Top < PictureBox3.Top Then
    If PictureBox1.Left - 10 < PictureBox3.Left And PictureBox1.Left + 70 > PictureBox3.Left Then
      PictureBox1.Top = -100 'ハートを上に移動
      PictureBox1.Left = Rnd() * (Me.Width - PictureBox1.Width) 'ハートの横の位置を決定
      a = 5 * Rnd() - 2
      Label1.Text = Label1.Text + 1 '点数を増やす
      PictureBox3.Top = -100 '矢を飛び去った状態にする
    
```

End If

End If

End If

If PictureBox3.Top < -70 Then 'もし、矢が飛び去ってしまったら

m = 0 '矢を非発射状態にする

End If

End Sub

Private Sub Form1_KeyDown(...省略...) Handles Me.KeyDown

```

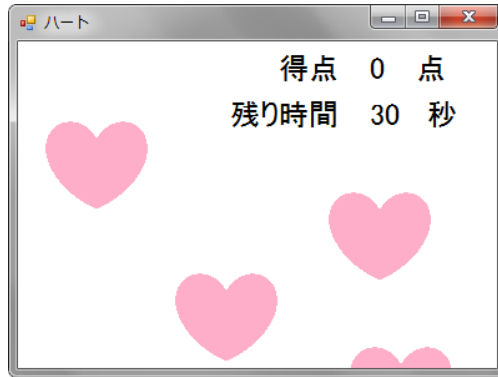
  If e.KeyCode = Keys.Left Then PictureBox2.Left = PictureBox2.Left - 3
  If e.KeyCode = Keys.Right Then PictureBox2.Left = PictureBox2.Left + 3
  If e.KeyCode = Keys.Up Then PictureBox2.Top = PictureBox2.Top - 3
  If e.KeyCode = Keys.Down Then PictureBox2.Top = PictureBox2.Top + 3
  If e.KeyCode = Keys.Space Then
    'もし、スペースキーをが押されたら
    If m = 0 Then
      'もし、矢が非発射状態なら
      m = 1
      '矢を発射状態にする
      PictureBox3.Top = PictureBox2.Top
      PictureBox3.Left = PictureBox2.Left + 52
    
```

```
End If
End If
End Sub

Private Sub Form1_Load(...省略...) Handles MyBase.Load
    Randomize()           '乱数系列の初期化
    PictureBox3.Top = -100 '矢を見えない位置に移動
    a = 5 * Rnd() - 2
End Sub
```

練習問題10

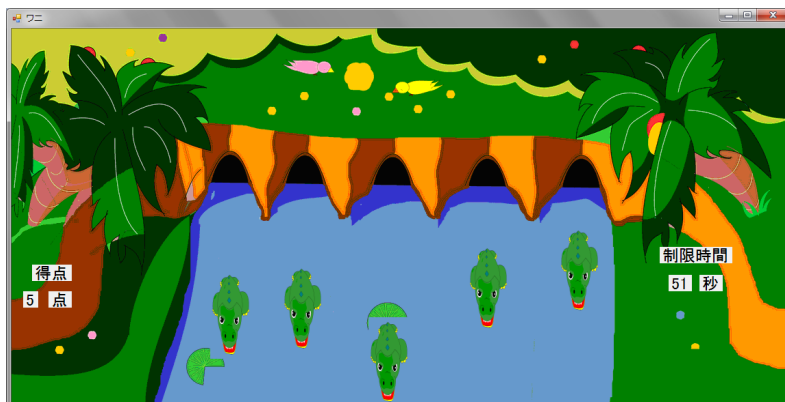
【1】例題10.3のハートの数を増やしなさい。また、制限時間も付けなさい。



【2】ハートが右から左へ飛んでくるように例題10.5のプログラムを改良しなさい。また、ハートの数も増やし制限時間も付けなさい。



【3】せまってくるワニをクリックして撃退するゲームを作りなさい。ワニをクリックすると得点が増えていきクリックされたワニはもと位置にもどります。フォームの下のラインまでワニが進むと噛まれたことになり減点されるようにしなさい。また、制限時間も付けなさい。



附錄



