

リスト2-2 2次元衝突と運動量保存の法則(つづき)

```

CY = CenterY + 200
Dim Pen1 As New Pen(Color.Red, 3) '幅3ピクセルのベクトル用
Dim Pen2 As New Pen(Color.Lime, 3)
Dim Pen3 As New Pen(Color.Blue, 3)
Pen1.CustomEndCap = New System.Drawing.Drawing2D.AdjustableArrowCap(4, 4) '矢印
Pen2.CustomEndCap = New System.Drawing.Drawing2D.AdjustableArrowCap(4, 4)
Pen3.CustomEndCap = New System.Drawing.Drawing2D.AdjustableArrowCap(4, 4)
gr.DrawLine(Pen1, CenterX - pxF / 2, CY + pyF / 2, CenterX - m2 * u / 2, CY + m2 * v2yF / 2)
gr.DrawLine(Pen2, CenterX - pxF / 2, CY + pyF / 2, CenterX - m1 * u / 2, CY + m1 * v1yF / 2)
gr.DrawLine(Pen3, CenterX - pxF / 2, CY + pyF / 2, CenterX, CY)
Dim Pen4 As New Pen(Color.Black, 1)
Pen4.DashStyle = Drawing2D.DashStyle.Dash '破線
gr.DrawLine(Pen4, CenterX - m2 * u / 2, CY + m2 * v2yF / 2, CenterX, CY) '補助線
gr.DrawLine(Pen4, CenterX - m1 * u / 2, CY + m1 * v1yF / 2, CenterX, CY)
'衝突後の2球の運動量のベクトル合成
gr.DrawLine(Pen1, CenterX, CY, CenterX + m1 * u / 2, CY - m1 * v1yL / 2)
gr.DrawLine(Pen2, CenterX, CY, CenterX + m2 * u / 2, CY - m2 * v2yL / 2)
gr.DrawLine(Pen3, CenterX, CY, CenterX + pxL / 2, CY - pyL / 2)
gr.DrawLine(Pen4, CenterX + m1 * u / 2, CY - m1 * v1yL / 2, CenterX + pxL / 2, CY - pyL / 2)
gr.DrawLine(Pen4, CenterX + m2 * u / 2, CY - m2 * v2yL / 2, CenterX + pxL / 2, CY - pyL / 2)
End If
End Sub

Private Sub Zahyo()
Dim gr As Graphics = PictureBox1.CreateGraphics()
Dim Pen As New Pen(Color.Black, 1)
Pen.DashStyle = Drawing2D.DashStyle.Dash
gr.DrawLine(Pen, CenterX - x0 - 30, CenterY, CenterX + x0 + 30, CenterY) 'X軸
gr.DrawLine(Pen, CenterX, 20, CenterX, 2 * CenterY - 20) 'Y軸
x1F = CenterX - x0 : y1F = CenterY - d1 - r1 '球1の初期位置
x2F = CenterX - x0 : y2F = CenterY + d2 + r2 '球2の初期位置
xGF = (m1 * x1F + m2 * x2F) / (m1 + m2) '系の重心のx座標
yGF = (m1 * y1F + m2 * y2F) / (m1 + m2) '系の重心のy座標
gr.FillEllipse(Brushes.Red, x1F - r1, y1F - r1, 2 * r1, 2 * r1) '初期の球1の描画
gr.FillEllipse(Brushes.Black, x1F - 2, y1F - 2, 4, 4)
gr.DrawEllipse(Pens.Black, x1F - r1, y1F - r1, 2 * r1, 2 * r1)
gr.FillEllipse(Brushes.Lime, x2F - r2, y2F - r2, 2 * r2, 2 * r2) '初期の球2の描画
gr.FillEllipse(Brushes.Black, x2F - 2, y2F - 2, 4, 4)
gr.DrawEllipse(Pens.Black, x2F - r2, y2F - r2, 2 * r2, 2 * r2)
gr.FillEllipse(Brushes.Black, xGF - 3, yGF - 3, 6, 6) '初期の系の重心の表示
gr.DrawLine(Pen, x1F, y1F, x2F, y2F) '球1,2を結ぶ点線
Dim MyFont1 As New Font("MSPゴシック", 12, FontStyle.Regular)
gr.DrawString("運動量保存の法則", MyFont1, Brushes.Black, CenterX - 70, 2 * CenterY + 140)
Dim MyFont2 As New Font("MSPゴシック", 10, FontStyle.Regular)
gr.DrawString("X", MyFont2, Brushes.Black, 2 * CenterX - 45, CenterY - 20)
gr.DrawString("Y", MyFont2, Brushes.Black, CenterX + 5, 10)
End Sub
End Class
    
```

後、時間  $t-t_0$  の球1, 2と重心位置をO点を起点に計算し、 $dt=1s$  ごとに表示していく。また衝突前後の両球の運動量ベクトルおよび合成ベクトルを描き、運動量の和が保存されていることがわかる。球1の運動量ベクトルを Pen1(赤)、球2の運動量ベクトルを Pen2(lime = 緑)、合成ベクトルを Pen3(青)の共に太さ3ピクセルの矢印(節1-5の図1-6およびカラー口絵i頁の③参照)で描画する。

2-3 | 自由落下運動の軌跡と力学的エネルギー保存 図2-5

● 落体の力学的エネルギーの保存

図2-6のように、真空中で  $h_0$  の高さから物体を自由落下(初速0)するものとする。落下後時間  $t$  における物体の速さ  $v$ 、落下距離  $y$  は、重力加速度を  $g$  として、



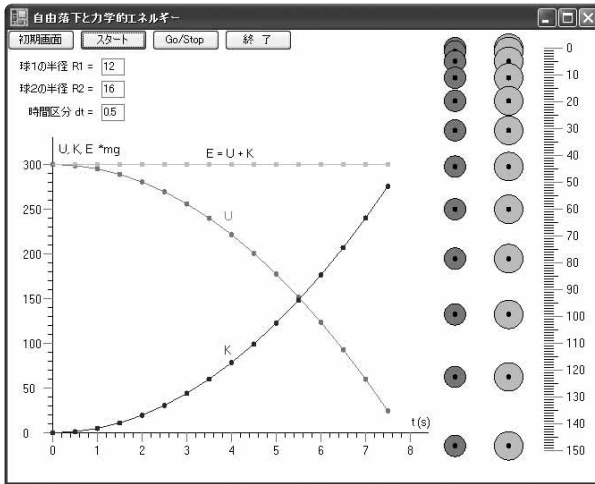


図2-5 リスト2-3の 実行画面例

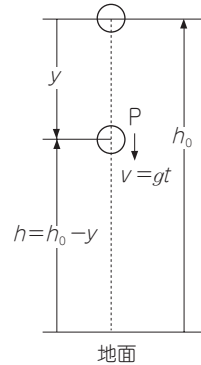


図2-6 自由落下運動

$$v = gt, \quad y = \frac{1}{2}gt^2 \quad \dots\dots\dots (2-21)$$

となる。「仕事・運動エネルギーの関係」によると、この時間  $t$  の間に重力がする仕事が運動エネルギーに等しいから、

$$mgy = \frac{1}{2}mv^2 \quad \dots\dots\dots (2-22)$$

が成り立つ。図の落体の位置 P 点の地上からの高さ  $h$  は、

$$h = h_0 - y \quad \dots\dots\dots (2-23)$$

であるから、これを用いて式(2-22)を書き直すと、

$$mgh_0 = mgh + \frac{1}{2}mv^2 \quad \dots\dots\dots (2-24)$$

が得られる。これは、落体の最初の位置エネルギー  $U_0 = mgh_0$  が落下中の位置エネルギー  $U = mgh$  と運動エネルギー  $K = mv^2/2$  の和に等しいことを示し、**力学的エネルギー保存の法則**と呼ばれる。

落下中の位置エネルギー、運動エネルギー、力学的エネルギーの時間的な変化の様子や関係をグラフで調べるだけならば、式(2-24)の両辺を  $mg$  で割った式：

$$h_0 = h + v^2/(2g) \quad \dots\dots\dots (2-25)$$

で行ってもよい。そこで、落下中の速さ  $v$ 、落下距離  $y$  が式(2-21)により変化していくとき、

$$\text{重力による位置エネルギー } U \text{ を, } h = h_0 - y \quad (\times mg) \quad \dots\dots\dots (2-26)$$

$$\text{運動エネルギー } K \text{ を, } v^2/(2g) \quad (\times mg) \quad \dots\dots\dots (2-27)$$

$$\text{力学的エネルギー } E \text{ を, } h + v^2/(2g) = h_0 - y + v^2/(2g) \quad (\times mg) \quad \dots\dots\dots (2-28)$$

によって、グラフ化することにする。

● プログラムの解説(リスト 2-3)

▶ Timer1 プロシージャ

・力学的エネルギー保存のグラフ

時間( $t$ )軸、エネルギー( $U, K, E$ )軸の原点  $O$  をフォーム座標点(50, 450)にとると、時間や各エネル

## リスト 2-3 自由落下運動の軌跡と力学的エネルギーの保存

```
'自由落下と力学的エネルギー

Public Class Form1

    Dim t, dt, tF, tL, v, y, g, UF, UL, KF, KL, EF, EL, r1, r2 As Single
    Dim MyFont As New Font("MSPゴシック", 10, FontStyle.Regular)

    '初期画面
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        r1 = 12 : TextBox1.Text = r1           '球1の半径
        r2 = 16 : TextBox2.Text = r2           '球2の半径
        dt = 0.5 : TextBox3.Text = dt         '時間区分(s)
        g = 9.8
        t = 0 : v = 0 : y = 0                 '初期条件
        tF = 50 + 50 * t : UF = 150 + y
        KF = 450 - v ^ 2 / (2 * g) : EF = 150 + y - v ^ 2 / (2 * g)
        Zahyo()
    End Sub

    'スタート
    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
        Timer1.Interval = 100
        Timer1.Enabled = True
    End Sub

    '再スタート
    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button3.Click
        Dim gr As Graphics = CreateGraphics()
        gr.Clear(Color.White)                '画面クリア
        r1 = TextBox1.Text                    '球1の半径
        r2 = TextBox2.Text                    '球2の半径
        dt = TextBox3.Text                    '時間区分
        t = 0 : v = 0 : y = 0                 '初期条件
        tF = 50 + 50 * t : UF = 150 + y
        KF = 450 - v ^ 2 / (2 * g)
        EF = 150 + y - v ^ 2 / (2 * g)
        Zahyo()
        Timer1.Enabled = True
    End Sub

    '終了
    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button4.Click
        End
    End Sub

    Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Timer1.Tick
        Dim gr As Graphics = CreateGraphics()
        t = t + dt
        v = g * t
        y = 1 / 2 * g * t ^ 2
        '力学的エネルギー保存のグラフ
        tL = 50 + 50 * t
        UL = 150 + y
        KL = 450 - v ^ 2 / (2 * g)
        EL = 150 + y - v ^ 2 / (2 * g)
        gr.FillEllipse(Brushes.Red, tL - 3, UL - 3, 6, 6) '位置エネルギーのプロット
        gr.DrawLine(Pens.Red, tF, UF, tL, UL)
        gr.FillEllipse(Brushes.Blue, tL - 3, KL - 3, 6, 6) '運動エネルギーのプロット
        gr.DrawLine(Pens.Blue, tF, KF, tL, KL)
        gr.FillEllipse(Brushes.Lime, tL - 3, EL - 3, 6, 6) '力学的エネルギーのプロット
        gr.DrawLine(Pens.Lime, tF, EF, tL, EL)
        tF = tL : UF = UL : KF = KL : EF = EL
        '自由落下運動の軌跡
        If y <= 150 Then
            gr.FillEllipse(Brushes.Red, 500 - r1, 20 + 3 * y - r1, 2 * r1, 2 * r1) '球1
```