

25.1.12 Projekt 2 – Bézier-Kurven

Für dieses Projekt gelten die gleichen Überlegungen und Kommentare wie für das Projekt im → Kapitel 23.3.5.3 (Paint-Projekt). Daher werden für die drei Beispiele nur der Quelltext und die Ergebnisse vorgestellt.

Bézier-Kurven

25.1.12.1 Beispiel 1

Im ersten Beispiel wird eine Bézier-Kurve dritten Grades gezeichnet:

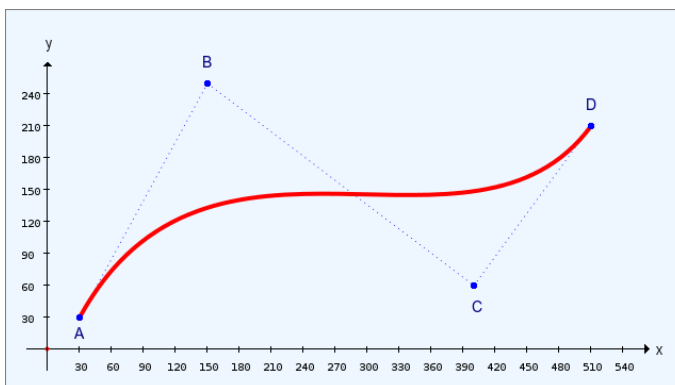


Abbildung 25.1.12.1.1: Modellierte Bézier-Kurve 3. Grades (4 Punkte)

Die Punkte A und D bestimmen den Anfangs- und Endpunkt der Kurve und über die Lage der Punkte B und C legen Sie das Krümmungsverhalten fest. Hier ist der vollständige Quelltext, um das Bild in der → Abbildung 25.1.12.1.1 zu zeichnen:

```
[1] Public Sub CairoScriptBezier4Points()
[2]     Dim vP As Vector
[3]
[4]     GenerateNewImage()
[5]     SetImageBorder()
[6]     Cairo.Begin(hImage)
[7]     Cairo.Translate(xTranslate, yTranslate)
[8]     Cairo.Scale(xScale, yScale) ' +y ▲
[9]     DrawCoordinateSystem()
[10]
[11]     ' BLATTFEDER
[12]     ' Vektor mit 8 Elementen, reelle Zahlen (keine komplexen Zahlen)
[13]     vP = New Vector(8, False)
[14]     ' A(30|30), B(150|250), C(400|60), D(510|210)
[15]     vP = [30, 30, 150, 250, 400, 60, 510, 210]
[16]
[17]     ' Verbindungslinien zeichnen A - B - C - D
[18]     Cairo.AntiAlias = 0
[19]     Cairo.Source = Cairo.ColorPattern(Color.Blue)
[20]     Cairo.LineWidth = 0.75
[21]     Cairo.Dash = [1, 4] ' Punktierte Linie ein
[22]     Cairo.MoveTo(vP[0], vP[1])
[23]     Cairo.LineTo(vP[2], vP[3])
[24]     Cairo.MoveTo(vP[2], vP[3])
[25]     Cairo.LineTo(vP[4], vP[5])
[26]     Cairo.MoveTo(vP[4], vP[5])
[27]     Cairo.LineTo(vP[6], vP[7])
[28]     Cairo.Stroke
[29]     Cairo.Dash = [] ' Punktierte Linie aus
```

```

[30]
[31] ' Bézier-Kurve
[32]   Cairo.Source = Cairo.ColorPattern(Color.Red)
[33]   Cairo.LineWidth = 4
[34]   Cairo.LineCap = Cairo.LineCapRound
[35]   Cairo.MoveTo(vP[0], vP[1])
[36]   Cairo.CurveTo(vP[2], vP[3], vP[4], vP[5], vP[6], vP[7])
[37]   Cairo.Stroke
[38]   Cairo.AntiAlias = 1
[39]
[40] ' Punkte A,B,C und D einzeichnen
[41]   Cairo.Source = Cairo.ColorPattern(Color.Blue)
[42]   Cairo.MoveTo(vP[0], vP[1])
[43]   Cairo.Arc(vP[0], vP[1], 3)
[44]   Cairo.MoveTo(vP[2], vP[3])
[45]   Cairo.Arc(vP[2], vP[3], 3)
[46]   Cairo.MoveTo(vP[4], vP[5])
[47]   Cairo.Arc(vP[4], vP[5], 3)
[48]   Cairo.MoveTo(vP[6], vP[7])
[49]   Cairo.Arc(vP[6], vP[7], 3)
[50]   Cairo.Fill
[51]
[52] ' TEXTE zeichnen
[53]   Cairo.NewPath
[54]   Cairo.Scale(1, -1)
[55]   Cairo.Font.Name = "Arial"
[56]   Cairo.Font.Size = 14
[57]   Cairo.Source = Cairo.ColorPattern(Color.DarkBlue)
[58]
[59]   Cairo.MoveTo(25, -10)
[60]   Cairo.DrawText("A")
[61]   Cairo.MoveTo(145, -265)
[62]   Cairo.DrawText("B")
[63]   Cairo.MoveTo(398, -35)
[64]   Cairo.DrawText("C")
[65]   Cairo.MoveTo(505, -225)
[66]   Cairo.DrawText("D")
[67]
[68]   Cairo.Scale(1, -1)
[69]   Cairo.End
[70]
[71] End ' CairoScriptBezier4Points()

```

Nur zur *Verdeutlichung der Lage aller Punkte in Bezug auf die Bézier-Kurve* werden in den Anweisungen ab Zeile 41 bestimmte Verbindungslinien und die Stützpunkte B und C eingezeichnet. Die Punkte A, B, C und D werden dabei als Kreis mit sehr kleinem Radius gezeichnet. Abschließend werden die vier Punkte noch bezeichnet – jedoch ohne die Angabe von Koordinaten.

25.1.12.2 Beispiel 2

In diesem Beispiel wird eine Zahl 2 (Ziffer-Symbol) aus drei Bézier-Kurven gezeichnet – aus zwei Bézier-Kurven 3. Grades und einer Strecke (Bézier-Kurven ersten Grades).

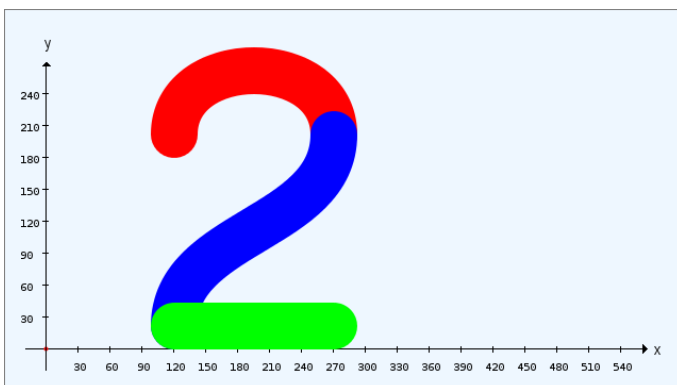


Abbildung 25.1.12.2.1: Die Zahl 2 aus drei Bézier-Kurven

Der Quelltext für die Prozedur *CairoScriptBezierNumber2()* wird vollständig angegeben:

```

[1] Public Sub CairoScriptBezierNumber2()
[2]   Dim PS, P1, P2, PE As New PointF
[3]   Dim vP, aQ As Vector
[4]   Dim fYOffset As Float

```

```

[5]
[6] GenerateNewImage()
[7] SetImageBorder()
[8] Cairo.Begin(hImage)
[9] Cairo.Translate(xTranslate, yTranslate)
[10] Cairo.Scale(xScale, yScale) ' +y ▲
[11] DrawCoordinateSystem()
[12]
[13] ' Ziffer 2 zeichnen
[14] Cairo.Scale(2, 2) ' +Zoom mit dem Faktor 2
[15] Cairo.LineWidth = 22
[16] fYOffset = Cairo.LineWidth / 2
[17] ' -----
[18] ' Oberer Bogen - rot
[19] PS.x = 60
[20] PS.y = 90 + fYOffset
[21] P1.x = 60
[22] P1.y = 130 + fYOffset
[23] P2.x = 135
[24] P2.y = 130 + fYOffset
[25] PE.x = 135
[26] PE.y = 90 + fYOffset
[27] Cairo.LineCap = Cairo.LineCapRound
[28] Cairo.Source = Cairo.ColorPattern(Color.Red)
[29] Cairo.MoveTo(PS.x, PS.y)
[30] Cairo.CurveTo(P1.x, P1.y, P2.x, P2.y, PE.x, PE.y)
[31] Cairo.Stroke
[32] ' -----
[33] ' Mittel-Kurve - blau
[34] vP = New Vector
[35] vP = [60, 0, 60, 45, 135, 45, 135, 90]
[36] aQ = New Vector
[37] aQ = [0, 1, 0, 1, 0, 1, 0, 1]
[38]
[39] vP = vP + fYOffset * aQ
[40] Cairo.Source = Cairo.ColorPattern(Color.Blue)
[41] Cairo.MoveTo(vP[0], vP[1])
[42] Cairo.CurveTo(vP[2], vP[3], vP[4], vP[5], vP[6], vP[7])
[43] Cairo.Stroke
[44] ' -----
[45] ' Untere Kurve - grün - Bézier-Kurve 1. Ordnung (Gerade)
[46] Cairo.MoveTo(60, 0 + fYOffset)
[47] Cairo.Source = Cairo.ColorPattern(Color.Green)
[48] Cairo.LineTo(135, 0 + fYOffset)
[49] Cairo.Stroke
[50] Cairo.End
[51]
[52] End ' CairoScriptBezierNumber2()

```

25.1.12.3 Beispiel 3

Im diesem Beispiel wird eine Grafik gezeichnet, die aus mehreren Bézier-Kurven 3. Grades und anderen Formen (Strecken, Kreise) besteht. Um die einzelnen Kurvenbögen zu unterscheiden, kommen unterschiedliche Farben zum Einsatz.



Abbildung 25.1.12.3.1: Komplexe Grafik mit Bézier-Kurven

Der Quelltext für das dritte Beispiel wird hier vollständig angegeben:

```

[1] Public Sub CairoScriptBezierSplines()
[2]     Dim vP As Vector
[3]
[4]     GenerateNewImage()
[5]     SetImageBorder()

```

```

[6] Cairo.Begin(hImage)
[7] Cairo.Translate(xTranslate, yTranslate)
[8] Cairo.Scale(xScale, yScale) ' +y ▲
[9] DrawCoordinateSystem()
[10]
[11] ' AUTO
[12] ' -----
[13] ' Stoßstange vorn
[14] Cairo.Scale(1.3, 1.3) ' +Zoom mit dem Faktor 1.3
[15] vP = New Vector
[16] vP = [1.25, 0.75, 1, 0.75, 1, 0.75, 1, 1]
[17] vP = 40 * vP
[18] Cairo.Source = Cairo.ColorPattern(Color.Red)
[19] Cairo.MoveTo(vP[0], vP[1])
[20] Cairo.CurveTo(vP[2], vP[3], vP[4], vP[5], vP[6], vP[7])
[21] Cairo.Stroke
[22] ' -----
[23] ' Kühlerhaube
[24] vP = New Vector
[25] vP = [1, 1, 1, 2, 2.5, 2.2, 3.8, 2.2]
[26] vP = 40 * vP
[27] Cairo.Source = Cairo.ColorPattern(Color.Green)
[28] Cairo.MoveTo(vP[0], vP[1])
[29] Cairo.CurveTo(vP[2], vP[3], vP[4], vP[5], vP[6], vP[7])
[30] Cairo.Stroke
[31] ' -----
[32] ' Frontscheibe und Dach
[33] vP = New Vector
[34] vP = [3.8, 2.2, 4, 3.8, 5.8, 3.2, 7.7, 2.6]
[35] vP = 40 * vP
[36] Cairo.Source = Cairo.ColorPattern(Color.Red)
[37] Cairo.MoveTo(vP[0], vP[1])
[38] Cairo.CurveTo(vP[2], vP[3], vP[4], vP[5], vP[6], vP[7])
[39] Cairo.Stroke
[40] ' -----
[41] ' Kofferraum-Abdeckung
[42] vP = New Vector
[43] vP = [7.7, 2.6, 8.5, 2.3, 8.6, 1.5, 10, 1]
[44] vP = 40 * vP
[45] Cairo.Source = Cairo.ColorPattern(Color.Blue)
[46] Cairo.MoveTo(vP[0], vP[1])
[47] Cairo.CurveTo(vP[2], vP[3], vP[4], vP[5], vP[6], vP[7])
[48] Cairo.Stroke
[49] ' -----
[50] ' Stoßstange hinten
[51] vP = New Vector
[52] vP = [10, 1, 10.5, 0.75, 10.5, 0.75, 10, 0.75]
[53] vP = 40 * vP
[54] Cairo.Source = Cairo.ColorPattern(Color.Red)
[55] Cairo.MoveTo(vP[0], vP[1])
[56] Cairo.CurveTo(vP[2], vP[3], vP[4], vP[5], vP[6], vP[7])
[57] Cairo.Stroke
[58] ' -----
[59] ' Bodenblech
[60] Cairo.AntiAlias = 1
[61] Cairo.MoveTo(400, 30)
[62] Cairo.LineWidth = 1
[63] Cairo.Source = Cairo.ColorPattern(Color.DarkGray)
[64] Cairo.LineTo(50, 30)
[65] Cairo.Stroke
[66] Cairo.AntiAlias = 0
[67] ' Rad vorn
[68] Cairo.Source = Cairo.ColorPattern(Color.DarkGreen)
[69] Cairo.Arc(100, 30, 30)
[70] Cairo.Fill
[71] Cairo.Source = Cairo.ColorPattern(Color.Gray)
[72] Cairo.Arc(100, 30, 17)
[73] Cairo.Fill
[74] ' Rad hinten
[75] Cairo.Source = Cairo.ColorPattern(Color.DarkGreen)
[76] Cairo.Arc(300, 30, 30)
[77] Cairo.Fill
[78] Cairo.Source = Cairo.ColorPattern(Color.Gray)
[79] Cairo.Arc(300, 30, 17)
[80] Cairo.Fill
[81]
[82] ' TEXT
[83] Cairo.NewPath
[84] Cairo.Scale(1 / 1.3, -1 / 1.3) ' ACHTUNG: y-Achse jetzt mit *positiven* Werten nach unten!
[85] Cairo.MoveTo(10, -220)
[86] Cairo.AntiAlias = 1
[87] Cairo.Source = Cairo.ColorPattern(Color.Gray)
[88] Cairo.LineTo(540, -220)
[89] Cairo.Stroke

```

```
[90] Cairo.AntiAlias = 0
[91] Cairo.Font.Name = "Monospace"
[92] Cairo.Font.Size = 30
[93] Cairo.MoveTo(50, -230)
[94] Cairo.DrawText("CAIRO-DESIGN")
[95] Cairo.End
[96]
[97] End ' CairoScriptBezierSplines()
```

Das Projekt-Archiv finden Sie im Download-Bereich.