Mathematica Code for “Curves and Surfaces for Computer Graphics”

Chapter 1

```
(* non-barycentric weights example *)
Clear[p0,p1,g1,g2,g3,g4];
p0={0,0}; p1={5,6};
g1=ParametricPlot[(1-t)^3 p0+t^3 p1,{t,0,1},PlotRange->All, Compiled->False,
 DisplayFunction->Identity];
g3=Graphics[{AbsolutePointSize[4], {Point[p0],Point[p1]} }];
p0={0,-1}; p1={5,5};
g2=ParametricPlot[(1-t)^3 p0+t^3 p1,{t,0,1},PlotRange->All, Compiled->False,
 PlotStyle->AbsoluteDashing[{2,2}], DisplayFunction->Identity];
g4=Graphics[{AbsolutePointSize[4], {Point[p0],Point[p1]} }];
Show[g2,g1,g3,g4, DisplayFunction->$DisplayFunction, DefaultFont->{"cmr10", 10}];
```

Figure 1.7, page 15: Effect of nonbarycentric Weights.

```
Clear[points];
points={{0,1},{1,1.1},{2,1.2},{3,3},{4,2.9},{5,2.8},{6,2.7}};
InterpolatingPolynomial[points,x];
Interpolation[points,InterpolationOrder->3];
Show[ListPlot[points,Prolog->AbsolutePointSize[5]],
 Plot[%%,{x,0,6},PlotStyle->Dashing[{0.05,0.05}]],
 Plot[%[x],{x,0,6}]]
```

Figure 1.10, page 19: Polynomial and Spline Fit.

Chapter 2

```
(* a bilinear surface patch *)
Clear[bilinear,pnts,u,w];
<<:Graphics:ParametricPlot3D.m;
pnts=ReadList["Points",{Number,Number,Number}, RecordLists->True];
bilinear[u_,w_]:=pnts[[1,1]](1-u)(1-w)+pnts[[1,2]]u(1-w) \
 +pnts[[2,1]]w(1-u)+pnts[[2,2]]u w;
Simplify[bilinear[u,w]]
g1=Graphics3D[{AbsolutePointSize[5], Table[Point[pnts[[i,j]]],{i,1,2},{j,1,2}]}];
g2=ParametricPlot3D[bilinear[u,w],{u,0,1,.05},{w,0,1,.05}, Compiled->False,
 DisplayFunction->Identity];
Show[g1,g2, ViewPoint->{0.063, -1.734, 2.905}];
{{0, 0, 1}, {1, 1, 1}, {1, 0, 0}, {0, 1, 0}}
{u + w - 2 u w, u, 1 - w}
```

Figure 2.7, page 61: A Bilinear Surface.

```
(* Another bilinear surface example *)
ParametricPlot3D[{0.5(1-u)w+u,w,(1-u)(1-w)}, {u,0,1},{w,0,1}, Compiled->False,
ViewPoint->{-0.846, -1.464, 3.997}, DefaultFont->{"cmr10", 10}];
```

Figure 2.8, page 62: A Bilinear Surface.

```
(* A Triangular bilinear surface example *)
```

```
ParametricPlot3D[{u(1-w),w,(1-u)(1-w)}, {u,0,1},{w,0,1}, Compiled->False,
ViewPoint->{-2.673, -3.418, 0.046}, DefaultFont->{"cmr10", 10}];
```

Figure 2.9, page 63, A Triangular Bilinear Surface.

```
Clear[loftedSurf]; (* double helix as a lofted surface *)
<<:Graphics:ParametricPlot3D.m;
loftedSurf:={Cos[u],Sin[u],u}(1-w)+{Cos[u+Pi],Sin[u+Pi],u}w;
ParametricPlot3D[loftedSurf, {u,0,Pi,.1},{w,0,1}, Compiled->False,
Ticks->False, ViewPoint->{-2.640, -0.129, 0.007}]
```

Figure 2.12, page 67, The Double Helix as a Lofted Surface.

```
(* Another lofted surface example *)
<<:Graphics:ParametricPlot3D.m
Clear[ls];
ls=Simplify[{8u^3-12u^2+6u-1,4u^3-9u^2+6u,0}(1-w)+{2u-1,4u(u-1),1}w];
ParametricPlot3D[ls, {u,0,1,.1},{w,0,1,.1}, Compiled->False,
 ViewPoint->{-0.139, -1.179, 1.475}, DefaultFont->{"cmr10", 10},
 AspectRatio->Automatic, Ticks->{{0,1},{0,1},{0,1}}];
```

Figure 2.13, page 68, A Lofted Surface Patch.

Chapter 3

```
(* 3-point Lagrange polynomial (uniform and nonunif) *)
Clear[T,H,B,d0,d1];
d0=1; d1=1;
T={t^2,t,1};
H={{1/(d0(d0+d1)),-1/(d0 d1),1/(d1(d0+d1))},
{-1/(d0+d1)-1/d0,1/d0+1/d1,-1/d1+1/(d0+d1)},{1,0,0}};
B={{1,0},{1.3,.5},{4,0}};
Simplify[T.H.B];
C1=ParametricPlot[T.H.B,{t,0,d0+d1},PlotRange->All, Compiled->False,
 PlotStyle->AbsoluteDashing[{2,2}], DisplayFunction->Identity];
d0=.583; d1=2.75;
H={{1/(d0(d0+d1)),-1/(d0 d1),1/(d1(d0+d1))},
{-1/(d0+d1)-1/d0,1/d0+1/d1,-1/d1+1/(d0+d1)},{1,0,0}};
Simplify[T.H.B];
C2=ParametricPlot[T.H.B,{t,0,d0+d1},PlotRange->All, Compiled->False,
 DisplayFunction->Identity];
Show[C1, C2, AspectRatio->Automatic, DefaultFont->{"cmr10", 10},
 DisplayFunction->$DisplayFunction];
```

Figure 3.1, page 80, Three-Point Lagrange Polynomials.

```
(* 3-point Lagrange polynomial (3 examples of nonuniform) *)
Clear[T,H,B,d0,d1,C1,C2,C3];
d0=1.414; d1=1.415; (* d1=0.5|P2-P1| *)
T={t^2,t,1};
H={{1/(d0(d0+d1)),-1/(d0 d1),1/(d1(d0+d1))},
{-1/(d0+d1)-1/d0,1/d0+1/d1,-1/d1+1/(d0+d1)},{1,0,0}};
B={{1,1},{2,2},{4,0}};
Simplify[T.H.B]
C1=ParametricPlot[T.H.B,{t,0,d0+d1},PlotRange->All, Compiled->False,
 DisplayFunction->Identity];
d1=2.83; (* d1=|P2-P1| *)
```

```
H={{1/(d0(d0+d1)),-1/(d0 d1),1/(d1(d0+d1))},
{-1/(d0+d1)-1/d0,1/d0+1/d1,-1/d1+1/(d0+d1)},{1,0,0}};
Simplify[T.H.B]
C2=ParametricPlot[T.H.B,{t,0,d0+d1},PlotRange->All, Compiled->False,
 DisplayFunction->Identity];
d1=5.66;  (* d1=2|P2-P1| *)
H={{1/(d0(d0+d1)),-1/(d0 d1),1/(d1(d0+d1))},
{-1/(d0+d1)-1/d0,1/d0+1/d1,-1/d1+1/(d0+d1)},{1,0,0}};
Simplify[T.H.B]
C3=ParametricPlot[T.H.B,{t,0,d0+d1},PlotRange->All, Compiled->False,
 DisplayFunction->Identity];
Show[C1,C2,C3, AspectRatio->Automatic, DefaultFont->{"cmr10", 10},
 DisplayFunction->$DisplayFunction];
(* (1/24,-1/8)t^3+(-1/3,3/4)t^2+(1,-1)t *)
```

Figure 3.2, page 81, Three-Point Nonuniform Lagrange Polynomials.

```
(* Plot quadratic and cubic Lagrange basis functions *)
lagq={t^2,t,1}.{{1/2,-1,1/2},{-3/2,2,-1/2},{1,0,0}};
Plot[{lagq[[1]],lagq[[2]],lagq[[3]]},{t,0,2},
 PlotRange->All, AspectRatio->Automatic, DefaultFont->{"cmr10", 10}];
lagc={t^3,t^2,t,1}.{{-9/2,27/2,-27/2,9/2},{9,-45/2,18,-9/2},
 2{-11/2,9,-9/2,1},{1,0,0,0}}
Plot[{lagc[[1]],lagc[[2]],lagc[[3]],lagc[[4]]},{t,0,1},
 PlotRange->All, AspectRatio->Automatic, DefaultFont->{"cmr10", 10}];
```

Figure 3.3, page 83, (a) Quadratic and (b) Cubic Lagrange Basis Functions.

```
<<:Graphics:ParametricPlot3D.m; (* Biquadratic patch for 9 points *)
Clear[T,pnt,M,g1,g2];
T[t_]:={t^2,t,1};
pnt={{{0,0,0},{1,0,0},{2,0,0}}, {{0,1,0},{1,1,1},{2,1,-.5}},
 {{0,2,0},{1,2,0},{2,2,0}}};
M={{2,-4,2},{-3,4,-1},{1,0,0}};
g2=Graphics3D[{AbsolutePointSize[4],
Table[Point[pnt[[i,j]]],{i,1,3},{j,1,3}] }];
comb[i_]:=(T[u].M.pnt)[[i]](Transpose[M].T[w])[[i]];
g1=ParametricPlot3D[comb[1]+comb[2]+comb[3], {u,0,1},{w,0,1},
Compiled->False, DisplayFunction->Identity];
Show[g1,g2, ViewPoint->{1.391, -2.776, 0.304}, DefaultFont->{"cmr10", 10},
DisplayFunction->$DisplayFunction]
```

Figure 3.4, page 88, A Biquadratic Surface Patch Example.

```
<<:Graphics:ParametricPlot3D.m; (* BiCubic patch for 16 points *)
Clear[T,pnt,M,g1,g2];
T[t_]:={t^3,t^2,t,1};
pnt={{{0,0,0},{1,0,0},{2,0,0},{3,0,0}}, {{0,1,0},{1,1,1},{2,1,-.5},{3,1,0}},
 {{0,2,-.5},{1,2,0},{2,2,.5},{3,2,0}},{{0,3,0},{1,3,0},{2,3,0},{3,3,0}}};
M={{-4.5,13.5,-13.5,4.5},{9,-22.5,18,-4.5},{-5.5,9,-4.5,1},{1,0,0,0}};
g2=Graphics3D[{AbsolutePointSize[3],
Table[Point[pnt[[i,j]]],{i,1,4},{j,1,4}] }];
comb[i_]:=(T[u].M.pnt)[[i]](Transpose[M].T[w])[[i]];
g1=ParametricPlot3D[comb[1]+comb[2]+comb[3]+comb[4], {u,0,1},{w,0,1},
Compiled->False, DisplayFunction->Identity];
Show[g1,g2, ViewPoint->{2.752, -0.750, 1.265}, DefaultFont->{"cmr10", 10},
```

```
 (*  ViewPoint->{1.413, 2.605, 0.974}  for alt view *)
 DisplayFunction->$DisplayFunction]
```

Figure 3.6, page 91, A Bicubic Surface Patch Example.

```
Clear[p0,p1,p2,p3,basis,fourP,g0,g1,g2,g3,g4,g5];
p0[u_]:={u,0,Sin[Pi u]}; p1[u_]:={u,1+u/10,Sin[Pi(u+.1)]};
p2[u_]:={u,2,Sin[Pi(u+.2)]}; p3[u_]:={u,3+u/10,Sin[Pi(u+.3)]};
(* matrix 'basis' has dimensions 4x4x3 *)
basis:={{p0[0],p0[.33],p0[.67],p0[1]},{p1[0],p1[.33],p1[.67],p1[1]},
{p2[0],p2[.33],p2[.67],p2[1]},{p3[0],p3[.33],p3[.67],p3[1]}};
fourP:= (* basis matrix for a 4-point curve *)
{{-4.5,13.5,-13.5,4.5},{9,-22.5,18,-4.5},{-5.5,9,-4.5,1},{1,0,0,0}};
prt[i_]:= (* extracts component i from the 3rd dimen of 'basis' *)
basis[[Range[1,4],Range[1,4],i]];
coord[i_]:= (* calc. the 3 parametric components of the surface *)
{u^3,u^2,u,1}.fourP.prt[i].Transpose[fourP].{w^3,w^2,w,1};
g0=ParametricPlot3D[p0[u], {u,0,1}]
g1=ParametricPlot3D[p1[u], {u,0,1}]
g2=ParametricPlot3D[p2[u], {u,0,1}]
g3=ParametricPlot3D[p3[u], {u,0,1}]
g4=Graphics3D[{AbsolutePointSize[4],
Table[Point[basis[[i,j]]],{i,1,4},{j,1,4}]}];
g5=ParametricPlot3D[{coord[1],coord[2],coord[3]},
{u,0,1,.05},{w,0,1,.05}, DisplayFunction->Identity];
Show[g0,g1,g2,g3, ViewPoint->{-2.576, -1.365, 1.718},
 Ticks->False, DisplayFunction->$DisplayFunction]
Show[g4,g5, ViewPoint->{-2.576, -1.365, 1.718},
 DisplayFunction->$DisplayFunction]
```

Figure 3.7, page 94, A Four-Curve Surface.

```
<<:Graphics:ParametricPlot3D.m;
Clear[p00,p01,p10,p11,pu0,pu1,p0w,p1w];
p00:={0,0,0}; p01:={0,1,0};
p10:={1,0,0}; p11:={1,1,0};
pu0:={u,0,Sin[Pi u]};
pu1:={u,1,Sin[Pi u]};
p0w:={0,w,Sin[Pi w]};
p1w:={1,w,Sin[Pi w]};
Simplify[
{1-u,u}.{p0w,p1w}+{1-w,w}.{pu0,pu1}
-p00(1-u)(1-w)-p01(1-u)w
-p10(1-w)u-p11 u w]
ParametricPlot3D[%,
{u,0,1,.2},{w,0,1,.2},
```

```
PlotRange->All,
AspectRatio->Automatic,
RenderAll->False,
Ticks->{{1},{0,1},{0,1}},
Prolog->AbsoluteThickness[.4]]
```

Figure 3.8, page 96, A Coons Surface.

```
p00={-1,-1,0}; p01={-1,1,0}; p10={1,-1,0}; p11={1,1,0};
pnts={p00,p01,p10,p11,{1,-1/2,1/2},{1,1/2,-1/2},
 {0,-1,-1/2},{0,1,1/2}};
p0w[w_]:={-1,2w-1,0};
p1w[w_]:={1,(-4-w+27w^2-18w^3)/4,27(w-3w^2+2w^3)/4};
pu0[u_]:={2u-1,-1,2u^2-2u};
pu1[u_]:={2u-1,1,-2u^2+2u};
p[u_,w_]:=(1-u)p0w[w]+u p1w[w]+(1-w)pu0[u]+w pu1[u] \
 -p00(1-u)(1-w)-p01(1-u)
w-p10 u(1-w)-p11 u w;
g1=Graphics3D[{AbsolutePointSize[5], Table[Point[pnts[[i]]],
{i,1,8}]}];
g2=ParametricPlot3D[p[u,w], {u,0,1},{w,0,1}, Compiled->False,
Ticks->{{-1,1},{-1,1},{-1,1}}, DisplayFunction->Identity];
Show[g1,g2]
```

Figure 3.9, page 98, A Coons Surface Patch and Code.

```
<<:Graphics:ParametricPlot3D.m; (* Triangular Coons patch *)
Clear[T,pnt,M,g1,g2];
T[t_]:={1+2t^3-3t^2,3t^2-2t^3,1};
p00={0,0,0}; p10={2,0,0}; p11={1,1,0};
M={{-p00,-p11,{w,w,4w(1-w)}},{-p10,-p11,{2-w,w,4w(1-w)}},
 {{2u,0,4u(u-1)},p11,{0,0,0}}};
g2=Graphics3D[{AbsolutePointSize[3],Point[p00], Point[p10], Point[p11] }];
comb[i_]:=(T[u].M)[[i]] T[w][[i]];
g1=ParametricPlot3D[comb[1]+comb[2]+comb[3], {u,0,1},{w,0,1},
Compiled->False, DisplayFunction->Identity];
Show[g1,g2, ViewPoint->{2.933, 0.824, 0.673}, DefaultFont->{"cmr10", 10},
DisplayFunction->$DisplayFunction]
(*ViewPoint->{1.413, 2.605, 0.974} for alt view *)
```

Figure 3.14, page 105, A Triangular Coons Surface Patch Example.

```
b[u_,w_]:={0,1/2,1}(1-u)(1-w)+{1,1/2,1}(1-u)w
 +{0,3/2,1}(1-w)u+{1,3/2,1}u w;
H={{2,-2,1,1},{-3,3,-2,-1},{0,0,1,0},{1,0,0,0}};
lu0={u^3,u^2,u,1}.H.{{0,0,0},{0,1/2,1},{0,0,1},{0,1,0}};
lu1={u^3,u^2,u,1}.H.{{1,0,0},{1,1/2,1},{0,0,1},{0,1,0}};
l[u_,w_]:=lu0(1-w)+lu1 w;
fu0={u^3,u^2,u,1}.H.{{3/2,1/2,0},{1,1/2,1},{0,0,1},{-1,0,0}};
```

```
fu1={u^3,u^2,u,1}.H.{{3/2,3/2,0},{1,3/2,1},{0,0,1},{-1,0,0}};
f[u_,w_]:=fu0(1-w)+fu1 w;
cu0={u^3,u^2,u,1}.H.{{1,0,0},{3/2,1/2,0},{1,0,0},{0,1,0}};
cu1={1,1/2,1};
c0w={w^3,w^2,w,1}.H.{{1,0,0},{1,1/2,1},{0,0,1},{0,1,0}};
c1w={w^3,w^2,w,1}.H.{{3/2,1/2,0},{1,1/2,1},{0,0,1},{-1,0,0}};
c[u_,w_]:=(1-u)c0w+u c1w+(1-w)cu0+w cu1 \
 -(1-u)(1-w){1,0,0}-u(1-w){3/2,1/2,0}-w(1-u)cu1- u w cu1;
g1=ParametricPlot3D[b[u,w], {u,0,1},{w,0,1}]
g2=ParametricPlot3D[l[u,w], {u,0,1},{w,0,1}]
g3=ParametricPlot3D[f[u,w], {u,0,1},{w,0,1}]
g4=ParametricPlot3D[c[u,w], {u,0,1},{w,0,1}]
Show[g1,g2,g3,g4]
```

Figure 3.15, page 107, Bilinear, Lofted, and Coons Surface Patches.

Chapter 4

```
Clear[T,H,B]; (* Hermite Interpolation *)
T={t^3,t^2,t,1};
H={{2,-2,1,1},{-3,3,-2,-1},{0,0,1,0},{1,0,0,0}};
B={{0,0},{2,1},{1,1},{1,0}};
ParametricPlot[T.H.B,{t,0,1},PlotRange->All]
```

Code on page 107.

```
<<:Graphics:ParametricPlot3D.m; (* Hermite 3D example *)
Clear[T,H,B];
T={t^3,t^2,t,1};
H={{2,-2,1,1},{-3,3,-2,-1},{0,0,1,0},{1,0,0,0}};
B={{0,0,0},{1,1,1},{1,0,0},{0,1,0}};
ParametricPlot3D[T.H.B,{t,0,1}, Compiled->False,
 ViewPoint->{-0.846, -1.464, 3.997}, DefaultFont->{"cmr10", 10}];
(* ViewPoint->{3.119, -0.019, 0.054} alt view *)
```

Figure 4.4, page 118, A Hermite Curve Segment in Space.

```
Clear[T,H,B]; (* Nonuniform Hermite segments *)
T={t^3,t^2,t,1};
H={{2,-2,1,1},{-3,3,-2,-1},{0,0,1,0},{1,0,0,0}};
B[delta_]:={{0,0},{2,0},delta{2,1},delta{2,-1}};
g1=ParametricPlot[T.H.B[0.5],{t,0,1},Compiled->False,
 DisplayFunction->Identity];
g2=ParametricPlot[T.H.B[1],{t,0,1},Compiled->False,
 DisplayFunction->Identity];
g3=ParametricPlot[T.H.B[1.5],{t,0,1},Compiled->False,
 DisplayFunction->Identity];
Show[g1,g2,g3, DisplayFunction->$DisplayFunction, DefaultFont->{"cmr10", 10}]
```

Figure 4.5, page 120, Three Nonuniform Hermite Segments.

```
<<:Graphics:ParametricPlot3D.m; (* Two Ferguson patches *)
F1[t_]:=2t^3-3t^2+1; F2[t_]:=-2t^3+3t^2;
F3[t_]:=t^3-2t^2+t; F4[t_]:=t^3-t^2;
F[t_]:={F1[t],F2[t],F3[t],F4[t]};
p00={0,0,0}; p01={0,1,0}; pu00={1,0,1}; pw00={0,1,1}; pu01={1,0,1}; pw01={0,1,0};
p10={1,0,0}; p11={1,1,0}; pu10={1,0,-1}; pw10={0,1,0}; pu11={1,0,-1}; pw11={0,1,-1};
p20={2,0,0}; p21={2,1,0}; pu20={1,0,0}; pw20={0,1,0}; pu21={1,0,0}; pw21={0,1,0};
H={{p00,p01,pw00,pw01},{p10,p11,pw10,pw11},
 {pu00,pu01,{0,0,0},{0,0,0}},{pu10,pu11,{0,0,0},{0,0,0}}};
prt[i_]:=H[[Range[1,4],Range[1,4],i]];
g1=ParametricPlot3D[{F[u].prt[1].F[w],F[u].prt[2].F[w],F[u].prt[3].F[w]},
{u,0,.98,.05},{w,0,1,.05}, DisplayFunction->Identity];

H={{p10,p11,pw10,pw11},{p20,p21,pw20,pw21},
 {pu10,pu11,{0,0,0},{0,0,0}},{pu20,pu21,{0,0,0},{0,0,0}}};
g2=ParametricPlot3D[{F[u].prt[1].F[w],F[u].prt[2].F[w],F[u].prt[3].F[w]},
{u,0.05,1,.05},{w,0,1,.05}, DisplayFunction->Identity];

g3=Graphics3D[{AbsolutePointSize[4],
Point[p00],Point[p01],Point[p10],Point[p11],Point[p20],Point[p21]}];

Show[g1,g2,g3, ViewPoint->{0.322, 1.342, 0.506},
DefaultFont->{"cmr10", 10}, DisplayFunction->$DisplayFunction]
```

Figure 4.15, page 135, Two Ferguson Surface Patches.

Chapter 5

```
(* tilted helix as a periodic curve *)
ParametricPlot3D[{.05t+Cos[t],Sin[t],.1t}, {t,0,10Pi}, Compiled->False,
 Ticks->{{-1,0,1,2},{-1,0,1},{0,1,2,3}}, DefaultFont->{"cmr10", 10},
 PlotPoints->100]
```

Figure 5.3, page 151, A Tilted Helix as a Periodic Curve.

```
(* Nonuniform cubic spline example *)
C1:=ParametricPlot[{1/24,-1/8}t^3+{-1/3,3/4}t^2+{1,-1}t, {t,0,2},
 PlotRange->All, Compiled->False, DisplayFunction->Identity];
C2:=ParametricPlot[{-1/12,0}t^2+{1/6,1/2}t+{1,0}, {t,0,2},
 PlotRange->All, Compiled->False, DisplayFunction->Identity];
C3:=ParametricPlot[-{1/24,1/8}t^3+{-1/12,0}t^2+{-1/6,1/2}t+{1,1}, {t,0,2},
 PlotRange->All, Compiled->False, DisplayFunction->Identity];
Show[C1, C2, C3, PlotRange->All, AspectRatio->Automatic,
 DisplayFunction->$DisplayFunction, DefaultFont->{"cmr10", 10}];
```

Figure 5.5, page 156, A Nonuniform Cubic Spline Example.

```
(* quadratic spline example *)
C1:=ParametricPlot[{t,t^2-t}, {t,0,1}, DisplayFunction->Identity];
C2:=ParametricPlot[{-t^2+t+1,t}, {t,0,1}, DisplayFunction->Identity];
C3:=ParametricPlot[{-t+1,-t^2+t+1}, {t,0,1}, DisplayFunction->Identity];
C4=Graphics[{AbsolutePointSize[3],
 Point[{0,0}], Point[{1,0}], Point[{1,1}], Point[{0,1}] }];
Show[C1, C2, C3, C4, DisplayFunction->$DisplayFunction,
 DefaultFont->{"cmr10", 10}, AspectRatio->Automatic]
```

Figure 5.6, page 158, A Quadratic Spline Example.

```
(* Cardinal spline example *)
```

```
T={t^3,t^2,t,1};
H[s_]:={{-s,2-s,s-2,s},{2s,s-3,3-2s,-s},{-s,0,s,0},{0,1,0,0}};
B={{1,3},{2,0},{3,2},{2,3}};
s=3/6; (* T=0 *)
g1=ParametricPlot[T.H[s].B,{t,0,1},PlotRange->All, Compiled->False,
 DisplayFunction->Identity];
s=2/6; (* T=1/3 *)
g2=ParametricPlot[T.H[s].B,{t,0,1},PlotRange->All, Compiled->False,
 DisplayFunction->Identity];
s=1/6; (* T=2/3 *)
g3=ParametricPlot[T.H[s].B,{t,0,1},PlotRange->All, Compiled->False,
 DisplayFunction->Identity];
s=0; (* T=1 *)
g4=ParametricPlot[T.H[s].B,{t,0,1},PlotRange->All, Compiled->False,
 DisplayFunction->Identity];
g5=Graphics[{AbsolutePointSize[4], Table[Point[B[[i]]],{i,1,4}] }];
Show[g1,g2,g3,g4,g5, DefaultFont->{"cmr10", 10},
 DisplayFunction->$DisplayFunction]
```

Figure 5.8, page 164, A Cardinal Spline Example.

```
0 0 0  1  0 0 2   0 0 3 0 0
0 1 0 .5 .5 1 2.5 .5 0 3 1 0
0 2 0 .5 2.5 0 2.5 2.5 1 3 2 0
0 3 0  1 3  0 2  3  0 3 3 0

<<:Graphics:ParametricPlot3D.m;
Clear[Pt,Bm,CRpatch,g1,g2,g3];
Pt=ReadList["CRpoints",{Number,Number,Number},
 RecordLists->True];
Bm:={{-.5,1.5,-1.5,.5},{1,-2.5,2,-.5},
 {-.5,0,.5,0},{0,1,0,0}};
CRpatch[i_]:= (* 1st patch, rows 1-4 *)
{u^3,u^2,u,1}.Bm.Pt[[{1,2,3,4},{1,2,3,4},i]].
 Transpose[Bm].{w^3,w^2,w,1};
g1=Graphics3D[{AbsolutePointSize[4],
Table[Point[Pt[[i,j]]],{i,1,4},{j,1,4}]}];
g2=ParametricPlot3D[{CRpatch[1],CRpatch[2],CRpatch[3]},
{u,0,.98,.1},{w,0,1,.1}, DisplayFunction->Identity];
Show[g1,g2, ViewPoint->{-4.322, 0.242, 0.306},
 DisplayFunction->$DisplayFunction]
```

Figure 5.10, page 166, A Catmull–Rom Surface Patch.

```
0 0 0  1  0 0 2   0 0 3 0 0
0 1 0 .5 .5 1 2.5 .5 0 3 1 0
0 2 0 .5 2.5 0 2.5 2.5 1 3 2 0
0 3 0  1 3  0 2  3  0 3 3 0
0 4 0  1 4  0 2  4  0 3 4 0

<<:Graphics:ParametricPlot3D.m;
Clear[Pt,Bm,CRpatch,CRpatchM,g1,g2,g3];
Pt=ReadList["CRpoints",{Number,Number,Number},
 RecordLists->True];
Bm:={{-.5,1.5,-1.5,.5},{1,-2.5,2,-.5},
 {-.5,0,.5,0},{0,1,0,0}};
CRpatch[i_]:= (* 1st patch, rows 1-4 *)
{u^3,u^2,u,1}.Bm.Pt[[{1,2,3,4},{1,2,3,4},i]].
 Transpose[Bm].{w^3,w^2,w,1};
```

```
CRpatchM[i_]:= (* 2nd patch, rows 2-5 *)
{u^3,u^2,u,1}.Bm.Pt[[{2,3,4,5},{1,2,3,4},i]].
 Transpose[Bm].{w^3,w^2,w,1};
g1=Graphics3D[{AbsolutePointSize[4],
Table[Point[Pt[[i,j]]],{i,1,5},{j,1,4}]}];
g2=ParametricPlot3D[{CRpatch[1],CRpatch[2],CRpatch[3]},
{u,0,.98,.1},{w,0,1,.1}, DisplayFunction->Identity];
g3=ParametricPlot3D[{CRpatchM[1],CRpatchM[2],CRpatchM[3]},
{u,0,1,.1},{w,0,1,.1}, DisplayFunction->Identity];
Show[g1,g2,g3, ViewPoint->{-4.322, 0.242, 0.306},
DisplayFunction->$DisplayFunction]
```

Figure 5.11, page 167, Two Catmull–Rom Surface Patches.

```
(* A Catmull-Rom surface with tension *)
<<:Graphics:ParametricPlot3D.m;
Clear[Pt,Bm,CRpatch,g1,g2,s];
Pt={{{0,3,0},{1,3,0},{2,3,0},{3,3,0}}, {{0,2,0},{.1,2,.9},{2.9,2,.9},{3,2,0}},
 {{0,1,0},{.1,1,.9},{2.9,1,.9},{3,1,0}}, {{0,0,0},{1,0,0},{2,0,0},{3,0,0}}};
Bm:={{-s,2-s,s-2,s},{2s,s-3,3-2s,-s}, {-s,0,s,0},{0,1,0,0}};
CRpatch[i_]:= (* rows 1-4 *)
{u^3,u^2,u,1}.Bm.Pt[[{1,2,3,4},{1,2,3,4},i]]. Transpose[Bm].{w^3,w^2,w,1};
g1=Graphics3D[{AbsolutePointSize[2],
Table[Point[Pt[[i,j]]],{i,1,4},{j,1,4}]}];
s=.4;
g2=ParametricPlot3D[{CRpatch[1],CRpatch[2],CRpatch[3]},
{u,0,1},{w,0,1}, DisplayFunction->Identity];
Show[g1,g2, ViewPoint->{1.431, -4.097, 0.011},
 DisplayFunction->$DisplayFunction, PlotRange->All]
```

Figure 5.12, page 168, A Catmull–Rom Surface Patch With Tension.

```
Clear[T,H,B,pts,Pa,Pd,te,bi,co]; (* Kochanek Bartels 3+2 points*)
T={t^3,t^2,t,1};
H={{2,-2,1,1},{-3,3,-2,-1},{0,0,1,0},{1,0,0,0}};
Pd[k_]:=(1-te[[k+1]])(1+bi[[k+1]])(1+co[[k+1]])(pts[[k+1]]-pts[[k]])/2+
 (1-te[[k+1]])(1-bi[[k+1]])(1-co[[k+1]])(pts[[k+2]]-pts[[k+1]])/2;
Pa[k_]:=(1-te[[k+2]])(1+bi[[k+2]])(1-co[[k+2]])(pts[[k+2]]-pts[[k+1]])/2+
 (1-te[[k+2]])(1-bi[[k+2]])(1+co[[k+2]])(pts[[k+3]]-pts[[k+2]])/2;
pts:={{-1,-1},{0,0},{4,6},{10,-1},{11,-2}};
te={0,0,0,0,0}; bi={0,0,0,0,0}; co={0,0,0,0,0};

B={pts[[2]],pts[[3]],Pd[1],Pa[1]};
Simplify[T.H.B]
Simplify[D[T.H.B,t]]
g1=ParametricPlot[T.H.B,{t,0,1},PlotRange->All];

B={pts[[3]],pts[[4]],Pd[2],Pa[2]};
Simplify[T.H.B]
Simplify[D[T.H.B,t]]
g2=ParametricPlot[T.H.B,{t,0,1},PlotRange->All];
g3=Graphics[{AbsolutePointSize[4], Table[Point[pts[[i]]],{i,1,5}] }];
Show[g1,g2,g3]
```

Code For Figure 5.17, page 173.

Chapter 6

```
(* Just the base functions bern. Note how "pwr" handles 0^0 *)
Clear[pwr,bern];
pwr[x_,y_]:=If[x==0 && y==0, 1, x^y];
bern[n_,i_,t_]:=Binomial[n,i]pwr[t,i]pwr[1-t,n-i] (* t^i x (1-t)^(n-i) *)
Plot[Evaluate[Table[bern[5,i,t], {i,0,5}]], {t,0,1}, DefaultFont->{"cmr10", 10}];
```

Figure 6.2, page 179, The Bernstein Polynomials for $n = 2, 3, 4$.

```
(* Just the base functions bern. Note how "pwr" handles 0^0 *)
Clear[pwr,bern,n,i,t]
pwr[x_,y_]:=If[x==0 && y==0, 1, x^y];
bern[n_,i_,t_]:=Binomial[n,i]pwr[t,i]pwr[1-t,n-i]
 (* t^i \[Times] (1-t)^(n-i) *)
Plot[Evaluate[Table[bern[5,i,t], {i,0,5}]], {t,0,1},
 DefaultFont->{"cmr10", 10}]
Clear[i,t,pnts,pwr,bern,bzCurve,g1,g2]; (* Cubic Bezier curve *)
(* either read points from file
pnts=ReadList["DataPoints",{Number,Number}]; *)
(* or enter them explicitly *)
pnts={{0,0},{.7,1},{.3,1},{1,0}}; (* 4 points for a cubic curve *)
pwr[x_,y_]:=If[x==0 && y==0, 1, x^y];
bern[n_,i_,t_]:=Binomial[n,i]pwr[t,i]pwr[1-t,n-i]
bzCurve[t_]:=Sum[pnts[[i+1]]bern[3,i,t], {i,0,3}]
g1=ListPlot[pnts, Prolog->AbsolutePointSize[4], PlotRange->All,
 AspectRatio->Automatic, DisplayFunction->Identity]
g2=ParametricPlot[bzCurve[t], {t,0,1}, DisplayFunction->Identity]
Show[g1,g2, DisplayFunction->$DisplayFunction]
```

Code on pages 179–180.

```
Clear[pnts,pwr,bern,bzCurve,g1,g2,g3]; (* General 3D Bezier curve *)
pnts={{1,0,0},{0,-3,0.5},{-3,0,0.75},{0,3,1},{3,0,1.5},{0,-3,1.75},{-1,0,2}};
n=Length[pnts]-1;
pwr[x_,y_]:=If[x==0 && y==0, 1, x^y];
bern[n_,i_,t_]:=Binomial[n,i]pwr[t,i]pwr[1-t,n-i] (* t^i x (1-t)^(n-i) *)
bzCurve[t_]:=Sum[pnts[[i+1]]bern[n,i,t], {i,0,n}];
g1=ParametricPlot3D[bzCurve[t], {t,0,1}, Compiled->False,
 DisplayFunction->Identity];
g2=Graphics3D[{AbsolutePointSize[2], Map[Point,pnts]}];
g3=Graphics3D[{AbsoluteThickness[2], (* control polygon *)
 Table[Line[{pnts[[j]],pnts[[j+1]]}], {j,1,n}]}];
g4=Graphics3D[{AbsoluteThickness[1.5], (* the coordinate axes *)
 Line[{{0,0,3},{0,0,0},{3,0,0},{0,0,0},{0,3,0}}]}];
Show[g1,g2,g3,g4, AspectRatio->Automatic, PlotRange->All, DefaultFont->{"cmr10", 10},
 Boxed->False, DisplayFunction->$DisplayFunction];
```

Code on page 180.

Q1:=$3\Delta t$;
Q2:=Q1$\times\Delta t$; // $3\Delta^2 t$
Q3:=$\Delta^3 t$;
Q4:=2Q2; // $6\Delta^2 t$
Q5:=6Q3; // $6\Delta^3 t$
Q6:=$\mathbf{P}_0 - 2\mathbf{P}_1 + \mathbf{P}_2$;
Q7:=$3(\mathbf{P}_1 - \mathbf{P}_2) - \mathbf{P}_0 + \mathbf{P}_3$;

```
B:=P0;
dB:=(P1 − P0)Q1+Q6×Q2+Q7×Q3;
ddB:=Q6×Q4+Q7×Q5;
dddB:=Q7×Q5;
for t:=0 to 1 step Δt do
Pixel(B);
B:=B+dB; dB:=dB+ddB; ddB:=ddB+dddB;
endfor;

n=3; Clear[q1,q2,q3,q4,q5,Q6,Q7,B,dB,ddB,dddB,p0,p1,p2,p3,tabl];
p0={0,1}; p1={5,.5}; p2={0,.5}; p3={0,1}; (* Four points *)
dt=.01; q1=3dt; q2=3dt^2; q3=dt^3; q4=2q2; q5=6q3;
Q6=p0-2p1+p2; Q7=3(p1-p2)-p0+p3;
B=p0; dB=(p1-p0) q1+Q6 q2+Q7 q3; (* space indicates *)
ddB=Q6 q4+Q7 q5; dddB=Q7 q5;     (* multiplication *)
tabl={};
Do[{tabl=Append[tabl,B], B=B+dB, dB=dB+ddB, ddB=ddB+dddB},
                                                  {t,0,1,dt}];
ListPlot[tabl];
```

Figure 6.4, page 189, A Fast Bézier Curve Algorithm.

```
(* New points for Bezier curve subdivision exercise *)
pnts={{0,1,1},{1,1,0},{4,2,0},{6,1,1}};
t=1/3;
pwr[x_,y_]:=If[x==0 && y==0, 1, x^y];
bern[n_,i_,t_]:=Binomial[n,i]pwr[t,i]pwr[1-t,n-i]
p01=Sum[pnts[[i+1]]bern[1,i,t], {i,0,1}]
p012=Sum[pnts[[i+1]]bern[2,i,t], {i,0,2}]
p0123=Sum[pnts[[i+1]]bern[3,i,t], {i,0,3}]
p0123=Sum[pnts[[3-3+i+1]]bern[3,i,t], {i,0,3}]
p123=Sum[pnts[[3-2+i+1]]bern[2,i,t], {i,0,2}]
p23=Sum[pnts[[3-1+i+1]]bern[1,i,t], {i,0,1}]
```

Figure Ans.10, page 423, Code to Compute Six New Points.

```
(* Interpolating Bezier Curve: I *)
Clear[p0,p1,p2,p3,p4,p5,x1,x2,x3,y1,y2,y3,c1,c2,c3,g1,g2,g3,g4];
p0={1/2,0}; p1={1/2,1/2}; p2={0,1};
 p3={1,3/2}; p4={3/2,1}; p5={1,1/2};
x1=p1+(p2-p0)/6;
x2=p2+(p3-p1)/6;
x3=p3+(p4-p2)/6;
y1=p2-(p3-p1)/6;
y2=p3-(p4-p2)/6;
y3=p4-(p5-p3)/6;
c1[t_]:=Simplify[(1-t)^3 p1+3t(1-t)^2 x1+3t^2(1-t) y1+t^3 p2]
```

```
c2[t_]:=Simplify[(1-t)^3 p2+3t(1-t)^2 x2+3t^2(1-t) y2+t^3 p3]
c3[t_]:=Simplify[(1-t)^3 p3+3t(1-t)^2 x3+3t^2(1-t) y3+t^3 p4]
g1=ListPlot[{p0,p1,p2,p3,p4,p5,x1,x2,x3,y1,y2,y3},
 Prolog->AbsolutePointSize[4], PlotRange->All,
 AspectRatio->Automatic, DisplayFunction->Identity]
g2=ParametricPlot[c1[t], {t,0,.9}, DisplayFunction->Identity]
g3=ParametricPlot[c2[t], {t,0.1,.9}, DisplayFunction->Identity]
g4=ParametricPlot[c3[t], {t,0.1,1}, DisplayFunction->Identity]
Show[g1,g2,g3,g4, DisplayFunction->$DisplayFunction]
```

Figure 6.18, page 215, An Interpolating Bézier Curve.

```
(* biquadratic bezier surface patch *)
Clear[pwr,bern,spnts,n,bzSurf,g1,g2];
n=2;
<<:Graphics:ParametricPlot3D.m
spnts={{{0,0,0},{1,0,1},{0,0,2}},
 {{1,1,0},{4,1,1},{1,1,2}}, {{0,2,0},{1,2,1},{0,2,2}}};
(* Handle Indeterminate condition *)
pwr[x_,y_]:=If[x==0 && y==0, 1, x^y];
bern[n_,i_,u_]:=Binomial[n,i]pwr[u,i]pwr[1-u,n-i]
bzSurf[u_,w_]:=Sum[bern[n,i,u] spnts[[i+1,j+1]] bern[n,j,w],
 {i,0,n}, {j,0,n}]
g1=ParametricPlot3D[bzSurf[u,w],{u,0,1}, {w,0,1},
 Ticks->{{0,1,4},{0,1,2},{0,1,2}},
 Compiled->False, DisplayFunction->Identity];
g2=Graphics3D[{AbsolutePointSize[3],
 Table[Point[spnts[[i,j]]],{i,1,n+1},{j,1,n+1}]}];
Show[g1,g2, ViewPoint->{2.783, -3.090, 1.243}, PlotRange->All,
 DefaultFont->{"cmr10", 10}, DisplayFunction->$DisplayFunction];
```

Figure 6.20, page 221, A Biquadratic Bézier Surface Patch.

```
(* A Bezier surface example. Given the six two-dimensional... *)
Clear[pnts,b1,b2,g1,g2,vlines,hlines];
pnts={{{0,1,0},{1,1,1},{2,1,0}},{{0,0,0},{1,0,0},{2,0,0}}};
b1[w_]:={1-w,w}; b2[u_]:={(1-u)^2,2u(1-u),u^2};
comb[i_]:=(b1[w].pnts)[[i]] b2[u][[i]];
g1=ParametricPlot3D[comb[1]+comb[2]+comb[3], {u,0,1},{w,0,1}, Compiled->False,
DefaultFont->{"cmr10", 10}, DisplayFunction->Identity,
 AspectRatio->Automatic, Ticks->{{0,1,2},{0,1},{0,.5}}];
g2=Graphics3D[{AbsolutePointSize[5],
 Table[Point[pnts[[i,j]]],{i,1,2},{j,1,3}]}];
vlines=Graphics3D[{AbsoluteThickness[2],
 Table[Line[{pnts[[1,j]],pnts[[2,j]]}], {j,1,3}]}];
hlines=Graphics3D[{AbsoluteThickness[2],
 Table[Line[{pnts[[i,j]],pnts[[i,j+1]]}], {i,1,2}, {j,1,2}]}];
Show[g1,g2,vlines,hlines, ViewPoint->{-0.139, -1.179, 1.475},
 DisplayFunction->$DisplayFunction, PlotRange->All, Shading->False,
 DefaultFont->{"cmr10", 10}];
```

Figure 6.21, page 223, A Lofted Bézier Surface Patch.

```
(* Degree elevation of a rect Bezier surface from 2x3 to 4x5 *)
Clear[p,q,r];
m=1; n=2;
p={{p00,p01,p02},{p10,p11,p12}}; (* array of points *)
```

```
r=Array[a, {m+3,n+3}]; (* extended array, still undefined *)
Part[r,1]=Table[a, {i,-1,m+2}];
Part[r,2]=Append[Prepend[Part[p,1],a],a];
Part[r,3]=Append[Prepend[Part[p,2],a],a];
Part[r,n+2]=Table[a, {i,-1,m+2}];
MatrixForm[r] (* display extended array *)
q[i_,j_]:=({i/(m+1),1-i/(m+1)}. (* dot product *)
 {{r[[i+1,j+1]],r[[i+1,j+2]]},{r[[i+2,j+1]],r[[i+2,j+2]]}}).
 {j/(n+1),1-j/(n+1)}
q[2,3] (* test *)
```

(a)

```
(* Degree elevation of a rect Bezier surface from 2x3 to 4x5 *)
Clear[p,r,comb];
m=1; n=2; (* set p to an array of 3D points *)
p={{{0,0,0},{1,0,1},{2,0,0}},{{0,1,0},{1,1,.5},{2,1,0}}};
r=Array[a, {m+3,n+3}]; (* extended array, still undefined *)
Part[r,1]=Table[{a,a,a}, {i,-1,m+2}];
Part[r,2]=Append[Prepend[Part[p,1],{a,a,a}],{a,a,a}];
Part[r,3]=Append[Prepend[Part[p,2],{a,a,a}],{a,a,a}];
Part[r,n+2]=Table[{a,a,a}, {i,-1,m+2}];
MatrixForm[r] (* display extended array *)
comb[i_,j_]:=({i/(m+1),1-i/(m+1)}.
{{r[[i+1,j+1]],r[[i+1,j+2]]},{r[[i+2,j+1]],r[[i+2,j+2]]}})[[1]]{j/(n+1),1-j/(n+1)}[[1]]+
({i/(m+1),1-i/(m+1)}.
{{r[[i+1,j+1]],r[[i+1,j+2]]},{r[[i+2,j+1]],r[[i+2,j+2]]}})[[2]]{j/(n+1),1-j/(n+1)}[[2]];
MatrixForm[Table[comb[i,j], {i,0,2},{j,0,3}]]
```

(b)

Figure 6.24, page 228, Code for Degree Elevation of a Rectangular Bézier Surface.

```
n=2; Clear[n,bern,p1,p2,g3,bzSurf,patch];
<<:Graphics:ParametricPlot3D.m
p1={{{-2,2,2},{-2,2,0},{0,2,0}},
 {{-4,0,2},{-4,0,0},{0,0,0}},
 {{-2,-2,2},{-2,-2,0},{0,-2,0}}};
p2={{{0,2,0},{2,2,0},{2,2,-2}},
 {{0,0,0},{4,0,0},{4,0,-2}},
 {{0,-2,0},{2,-2,0},{2,-2,-2}}};
pwr[x_,y_]:=If[x==0 && y==0, 1, x^y];
bern[n_,i_,u_]:=Binomial[n,i]pwr[u,i]pwr[1-u,n-i]
bzSurf[p_]:={Sum[p[[i+1,j+1,1]]bern[n,i,u]bern[n,j,w],
 {i,0,n,1}, {j,0,n,1}],
 Sum[p[[i+1,j+1,2]]bern[n,i,u]bern[n,j,w],
 {i,0,n,1}, {j,0,n,1}],
 Sum[p[[i+1,j+1,3]]bern[n,i,u]bern[n,j,w],
 {i,0,n,1}, {j,0,n,1}]};
patch[s_]:=
ParametricPlot3D[bzSurf[s],{u,0,1,.1}, {w,0.02,.98,.1}];
g3=Graphics3D[{AbsolutePointSize[3],
 Table[Point[p1[[i,j]]],{i,1,n+1},{j,1,n+1}]}]
```

```
g4=Graphics3D[{AbsolutePointSize[3],
 Table[Point[p2[[i,j]]],{i,1,n+1},{j,1,n+1}]}]
Show[patch[p1],patch[p2],g3,g4,
 DisplayFunction->$DisplayFunction]
```

Figure 6.26, page 231, Two Bézier Surface Patches.

```
(* A Rational Bezier Surface *)
Clear[pwr,bern,spnts,n,m,wt,bzSurf,cpnts,patch,vlines,hlines,axes];
<<:Graphics:ParametricPlot3D.m
spnts={{{0,0,0},{1,0,1},{0,0,2}},
 {{1,1,0},{4,1,1},{1,1,2}}, {{0,2,0},{1,2,1},{0,2,2}}};
m=Length[spnts[[1]]]-1; n=Length[Transpose[spnts][[1]]]-1;
wt=Table[1, {i,1,n+1},{j,1,m+1}];
wt[[2,2]]=5;
pwr[x_,y_]:=If[x==0 && y==0, 1, x^y];
bern[n_,i_,u_]:=Binomial[n,i]pwr[u,i]pwr[1-u,n-i]
bzSurf[u_,w_]:=
 Sum[wt[[i+1,j+1]]spnts[[i+1,j+1]]bern[n,i,u]bern[m,j,w], {i,0,n}, {j,0,m}]/
 Sum[wt[[i+1,j+1]]bern[n,i,u]bern[m,j,w], {i,0,n}, {j,0,m}];
patch=ParametricPlot3D[bzSurf[u,w],{u,0,1}, {w,0,1},
 Compiled->False, DisplayFunction->Identity];
cpnts=Graphics3D[{AbsolutePointSize[4], (* control points *)
 Table[Point[spnts[[i,j]]], {i,1,n+1},{j,1,m+1}]}];
vlines=Graphics3D[{AbsoluteThickness[1], (* control polygon *)
 Table[Line[{spnts[[i,j]],spnts[[i+1,j]]}], {i,1,n}, {j,1,m+1}]}];
hlines=Graphics3D[{AbsoluteThickness[1],
 Table[Line[{spnts[[i,j]],spnts[[i,j+1]]}], {i,1,n+1}, {j,1,m}]}];
maxx=Max[Flatten[Table[Part[spnts[[i,j]], 1], {i,1,n+1}, {j,1,m+1}]]];
maxy=Max[Flatten[Table[Part[spnts[[i,j]], 2], {i,1,n+1}, {j,1,m+1}]]];
maxz=Max[Flatten[Table[Part[spnts[[i,j]], 3], {i,1,n+1}, {j,1,m+1}]]];
axes=Graphics3D[{AbsoluteThickness[1.5], (* the coordinate axes *)
 Line[{{0,0,maxz},{0,0,0},{maxx,0,0},{0,0,0},{0,maxy,0}}]}];
Show[cpnts,hlines,vlines,axes,patch, PlotRange->All, DefaultFont->{"cmr10",10},
 DisplayFunction->$DisplayFunction, ViewPoint->{2.783, -3.090, 1.243}];
```

Figure 6.27, page 233, A Rational Bézier Surface Patch.

```
(* Triangular Bezier surface patch *)
pnts={{3,3,0}, {2,2,0},{4,2,1}, {1,1,0},{3,1,1},{5,1,2},
 {0,0,0},{2,0,1},{4,0,2},{6,0,3}};
B[i_,j_,k_]:=(n!/(i! j! k!))u^i v^j w^k;
n=3; u=1/6; v=2/6; w=3/6; Tsrpt={0,0,0};
indx:=(n-j)(n-j+1)/2+1+i;
Do[{k=n-i-j, Tsrpt=Tsrpt+B[i,j,k] pnts[[indx]]}, {j,0,n}, {i,0,n-j}];
Tsrpt
```

Figure 6.30, page 236, Code for One Point in a Triangular Bézier Patch.

```
(* Triangular Bezier patch by Garry Helzer *)
rules=Solve[{u{a1,b1}+v{a2,b2}+w{a3,b3}=={x,y},u+v+w==1},{u,v,w}]
BarycentricCoordinates[Polygon[{{a1_,b1_},{a2_,b2_},{a3_,b3_}}]] \
 [{x_,y_}]={u,v,w}/.rules//Flatten
Subdivide[l_]:=l/. Polygon[{p_,q_,r_}] :> Polygon /@ \
 ({{p+p,p+q,p+r},{p+q,q+q,q+r},{p+r,q+r,r+r},{p+q,q+r,r+p}}/2)
Transform[F_][L_]:= L /. Polygon[l_] :> Polygon[F /@ l]
P[L_][{u_,v_,w_}]:=
```

```
Module[{x,y,z,n=(Sqrt[8Length[L]+1]-3)/2},
 ((List @@ Expand[(x+y+z)^n]) /. {x->u,y->v,z->w}).L]
Param[T_,L_][{x_,y_}]:=With[{p=BarycentricCoordinates[T][{x, y}]},P[L][p]]
```

Run the code below in a separate cell

```
(* Triangular bezier patch for n=3 *)
T=Polygon[{{1, 0}, {0, 1}, {0, 0}}];
L={P300,P210,P120,P030, P201,P111,P021, P102,P012, P003} \
 ={{3,0,0},{2.5,1,.5},{2,2,0},{1.5,3,0},
  {2,0,1},{1.5,1,2},{1,2,.5}, {1,0,1},{.5,1,.5}, {0,0,0}};
SubT=Nest[Subdivide, T, 3];
Patch=Transform[Param[T, L]][SubT];
cpts={PointSize[0.02], Point/@L};
coord={AbsoluteThickness[1],
 Line/@{{{0,0,0},{3.2,0,0}},{{0,0,0},{0,3.4,0}},{{0,0,0},{0,0,1.3}}}};
cpolygon={AbsoluteThickness[2],
 Line[{P300,P210,P120,P030,P021,P012,P003,P102,P201,P300}],
 Line[{P012,P102,P111,P120,P021,P111,P201,P210,P111,P012}]};
Show[Graphics3D[{cpolygon,cpts,coord,Patch}], Boxed->False, PlotRange->All,
 ViewPoint->{2.620, -3.176, 2.236}];
```

Figure 6.31, page 237, A Triangular Bézier Surface Patch For $n = 3$.

```
B={{(1 - a)^3, 3*(-1 + a)^2*a, 3*(1 - a)*a^2, a^3},
  {(-1 + a)^2*(1 - b), (-1 + a)*(-2*a - b + 3*a*b),
   a*(a + 2*b - 3*a*b),
   a^2*b}, {(1 - a)*(-1 + b)^2, (-1 + b)*(-a - 2*b + 3*a*b),
   b*(2*a + b - 3*a*b), a*b^2},
  {(1 - b)^3, 3*(-1 + b)^2*b, 3*(1 - b)*b^2, b^3}};
TB={{(1 - c)^3, (-1 + c)^2*(1 - d), (1 - c)*(-1 + d)^2,
   (1 - d)^3},
  {3*(-1 + c)^2*c, (-1 + c)*(-2*c - d + 3*c*d),
  (-1 + d)*(-c - 2*d + 3*c*d), 3*(-1 + d)^2*d},
  {3*(1 - c)*c^2, c*(c + 2*d - 3*c*d), d*(2*c + d - 3*c*d),
  3*(1 - d)*d^2},
  {c^3, c^2*d, c*d^2, d^3}};
P={{P30,P31,P32,P33},{P20,P21,P22,P23},
 {P10,P11,P12,P13},{P00,P01,P02,P03}};
Q=Simplify[B.P.TB]
```

Code on page 248.

Chapter 7

```
(* B-spline example of 2 cubic segs and 3 quadr segs for 5 points *)
Clear[Pt,T,t,M3,comb,a,g1,g2,g3];
Pt={{0,0},{0,1},{1,1},{2,1},{2,0}};
(* first, 2 cubic segments (dashed) *)
T[t_]:={t^3,t^2,t,1};
M3={{-1,3,-3,1},{3,-6,3,0},{-3,0,3,0},{1,4,1,0}}/6;
```

```
comb[i_]:=(T[t].M3)[[i]] Pt[[i+a]];
g1=Graphics[{PointSize[.02], Point/@Pt}];
a=0;
g2=ParametricPlot[comb[1]+comb[2]+comb[3]+comb[4], {t,0,.95},
 Compiled->False, PlotRange->All, DisplayFunction->Identity,
 PlotStyle->AbsoluteDashing[{2,2}]];
a=1;
g3=ParametricPlot[comb[1]+comb[2]+comb[3]+comb[4], {t,0.05,1},
 Compiled->False, PlotRange->All, DisplayFunction->Identity,
 PlotStyle->AbsoluteDashing[{2,2}]];
(* Now the 3 quadratic segments (solid) *)
T[t_]:={t^2,t,1};
M2={{1,-2,1},{-2,2,0},{1,1,0}}/2;
comb[i_]:=(T[t].M2)[[i]] Pt[[i+a]];
a=0;
g4=ParametricPlot[comb[1]+comb[2]+comb[3], {t,0,.97},
 Compiled->False, PlotRange->All, DisplayFunction->Identity];
a=1;
g5=ParametricPlot[comb[1]+comb[2]+comb[3], {t,0.03,.97},
 Compiled->False, PlotRange->All, DisplayFunction->Identity];
a=2;
g6=ParametricPlot[comb[1]+comb[2]+comb[3], {t,0,1},
 Compiled->False, PlotRange->All, DisplayFunction->Identity];
Show[g2,g3,g4,g5,g6,g1, PlotRange->All, DefaultFont->{"cmr10", 10},
 DisplayFunction->$DisplayFunction];
```

Figure 7.4, page 260, Two Cubic (Dashed) and Three Quadratic (Solid) Segments of a B-spline.

```
(* Cubic B-spline with tension *)
Clear[t,s,pnts,stnp,tensMat,bsplineTensn,g1,g2,g3,g4];
pnts={{0,0},{0,1},{1,1},{1,0}};
stnp=Transpose[pnts];
tensMat={{2-s,6-s,s-6,s-2},{2s-3,s-9,9-2s,3-s},{-s,0,s,0},{1,4,1,0}};
bsplineTensn[t_]:=Module[{tmpstruc}, tmpstruc={t^3,t^2,t,1}.tensMat;
{tmpstruc.stnp[[1]],tmpstruc.stnp[[2]]}/6];
g1=ListPlot[pnts, Prolog->AbsolutePointSize[3],
 DisplayFunction->Identity];
s=0;
g2=ParametricPlot[bsplineTensn[t], {t,0,1},
 Compiled->False, DisplayFunction->Identity];
s=3;
g3=ParametricPlot[bsplineTensn[t], {t,0,1},
 Compiled->False, DisplayFunction->Identity,
 PlotStyle->AbsoluteDashing[{2,2}]];
s=5;
g4=ParametricPlot[bsplineTensn[t], {t,0,1},
 Compiled->False, DisplayFunction->Identity,
 PlotStyle->AbsoluteDashing[{1,2,2,2}]];
Show[g1,g2,g3,g4, DisplayFunction->$DisplayFunction]
```

Figure 7.7, page 267, Code for a Cubic B-Spline with Tension.

```
(* B-spline weight functions printed and plotted *)
Clear[bspl,knt,i,k,n,t,p]
bspl[i_,k_,t_]:=If[knt[[i+k]]==knt[[i+1]],0, (* 0<=i<=n *)
 bspl[i,k-1,t] (t-knt[[i+1]])/(knt[[i+k]]-knt[[i+1]])] \
 +If[knt[[i+1+k]]==knt[[i+2]],0,
```

```
 bspl[i+1,k-1,t] (knt[[i+1+k]]-t)/(knt[[i+1+k]]-knt[[i+2]])];
bspl[i_,1,t_]:=If[knt[[i+1]]<=t<knt[[i+2]], 1, 0];
n=4; k=3; (* Note: 0<=k<=n *)
(* knt=Table[i, {i,0,n+k}]; *) (* knots for the uniform case *)
knt={0,0,0,1,2,3,3,3}; (* knots for the NONuniform case *)
(* Show the weight functions *)
Do[Print["N(",i,",",k,",",t,")=",Simplify[bspl[i,k,t]]], {i,0,n}]
(* Plot them. Plots are separated using .97 instead of 1 *)
Do[p[i+1]=Plot[bspl[i,k,t], {t,k-.97,n+.97},
 DisplayFunction->Identity], {i,0,n}]
Show[Table[p[i+1], {i,0,n}], Ticks->None,
 DisplayFunction->$DisplayFunction]
```

Figure 7.16, page 281, Code for the B-Spline Weight Functions.

```
(* Plot a B-spline curve. Can also print the weight functions *)
Clear[bspl,knt,i,k,n,t,p,g1,g2,pnt] (* First the weight functions *)
bspl[i_,k_,t_]:=If[knt[[i+k]]==knt[[i+1]],0, (* 0<=i<=n *)
 bspl[i,k-1,t] (t-knt[[i+1]])/(knt[[i+k]]-knt[[i+1]])] \
 +If[knt[[i+1+k]]==knt[[i+2]],0,
 bspl[i+1,k-1,t] (knt[[i+1+k]]-t)/(knt[[i+1+k]]-knt[[i+2]])];
bspl[i_,1,t_]:=If[knt[[i+1]]<=t<knt[[i+2]], 1, 0];
n=4; k=3; (* Note: 0<=k<=n *)
(* knt=Table[i, {i,0,n+k}]; knots for the uniform case *)
knt={0,0,0,1,2,3,3,3}; (* knots for the open-unif or non-uniform cases *)
(* Do[Print[bspl[i,k,t]], {i,0,n}] Display the weight functions *)
pnt={{0,0},{1,1},{1,2},{2,2},{3,1}}; (* test for n+1=5 control points *)
p[t_]:=Sum[pnt[[i+1]] bspl[i,k,t], {i,0,n}] (* The curve as a weighted sum *)
g1=ListPlot[pnt, Prolog->AbsolutePointSize[3], DisplayFunction->Identity];
g2=ParametricPlot[p[t], {t,0,.97}, Compiled->False, DisplayFunction->Identity];
g3=ParametricPlot[p[t], {t,1,1.97}, Compiled->False, DisplayFunction->Identity];
g4=ParametricPlot[p[t], {t,2,3}, Compiled->False, DisplayFunction->Identity];
Show[g1,g2,g3,g4, PlotRange->All, DisplayFunction->$DisplayFunction,
 DefaultFont->{"cmr10", 10}];
```

Figure 7.17, page 283, An Open Uniform B-Spline.

```
(* 8-Point Nonuniform Cubic B-Spline Example. Five Segments *)
Clear[g,Q,pts,seg];
P0={0,0}; P1={0,1}; P2={1,1}; P3={1,0}; P4={2,0}; P5={2.75,1}; P6={3,1}; P7={3,0};
pts=Graphics[{PointSize[.01], Point/@{P0,P1,P2,P3,P4,P5,P6,P7}}];
seg={AbsoluteDashing[{2,2}], Line[{P1,P2,P3}], Line[{P4,P5,P6,P7}]};
Q[t_]:={((1-t)^3 P0 +(3t^3-6t^2+4) P1 +(-3t^3+3t^2+3t+1) P2 +t^3 P3)/6,
((2-t)^3 P1 +(3t^3-15t^2+21t-5) P2 +(-3t^3+12t^2-12t+4) P3 +(t-1)^3 P4)/6,
((3-t)^3 P2 +(3t^3-24t^2+60t-44) P3 +(-3t^3+21t^2-45t+31) P4 +(t-2)^3 P5)/6,
((4-t)^3 P3 +(3t^3-33t^2+117t-131) P4 +(-3t^3+30t^2-96t+100) P5 +(t-3)^3 P6)/6,
((5-t)^3 P4 +(3t^3-42t^2+192t-284) P5 +(-3t^3+39t^2-165t+229) P6 +(t-4)^3 P7)/6};
g=Table[ParametricPlot[Q[t][[i]], {t,i-1,0.97i},
 Compiled->False, DisplayFunction->Identity], {i,1,5}];
Show[g, pts, Graphics[seg], PlotRange->All, DefaultFont->{"cmr10", 10},
 DisplayFunction->$DisplayFunction, AspectRatio->Automatic];
```

For the four segments of part (b), the only difference is

```
Q[t_]:={(1-t)^3/6 P0 +(11t^3-15t^2-3t+7)/12 P1+(-5t^3+3t^2+3t+1)/4 P2 +t^3/2 P3,
(2-t)^3/2 P2 +(5t^3-27t^2+45t-21)/4 P3+(-11t^3+51t^2-69t+29)/12 P4 +(t-1)^3/6 P5,
(3-t)^3/4 P3 +(7t^3-57t^2+147t-115)/12 P4+(-3t^3+21t^2-45t+31)/6 P5 +(t-2)^3/6 P6,
((4-t)^3 P4 +(3t^3-33t^2+117t-131)P5+(-3t^3+30t^2-96t+100)P6 +(t-3)^3 P7)/6};
g=Table[ParametricPlot[Q[t][[i]], {t,i-1,0.97i},
 Compiled->False, DisplayFunction->Identity], {i,1,4}];
```

For the three segments of part (c), the only difference is

```
Q[t_]:={(1-t)^3 P0 /6+(11t^3-15t^2-3t+7)P1 /12+(-7t^3+3t^2+3t+1)P2 /4+t^3 P3,
(2-t)^3 P3+(7t^3-39t^2+69t-37)P4 /4+(-11t^3+51t^2-69t+29) P5 /12+(t-1)^3 P6 /6,
(3-t)^3 P4 /4+(7t^3-57t^2+147t-115)P5 /12+(-3t^3+21t^2-45t+31) P6 /6+(t-2)^3 P7 /6};
g=Table[ParametricPlot[Q[t][[i]], {t,i-1,0.97i},
 Compiled->False, DisplayFunction->Identity], {i,1,3}];
```

For the two segments of part (d), the only difference is

```
Q[t_]:={(1-t)^3P0 /6 +(11t^3-15t^2-3t+7)P1 /12+(-7t^3+3t^2+3t+1)P2 /4 +t^3 P3,
(2-t)^3 P4 +(7t^3-39t^2+69t-37)P5 /4+(-11t^3+51t^2-69t+29)P6 /12+(t-1)^3P7 /6};
g=Table[ParametricPlot[Q[t][[i]], {t,i-1,0.97i},
 Compiled->False, DisplayFunction->Identity], {i,1,2}];
```

Figure 7.18, page 287, Code for an 8-Point Nonuniform B-Spline Example, Figure 7.19.

```
(* Compute the nonuniform weight functions for the 8-point example that follows *)
Clear[bspl,knt]
bspl[i_,k_,t_]:=If[knt[[i+k]]==knt[[i+1]],0, (* 0<=i<=n *)
 bspl[i,k-1,t] (t-knt[[i+1]])/(knt[[i+k]]-knt[[i+1]])] \
 +If[knt[[i+1+k]]==knt[[i+2]],0,
 bspl[i+1,k-1,t] (knt[[i+1+k]]-t)/(knt[[i+1+k]]-knt[[i+2]])];
bspl[i_,1,t_]:=If[knt[[i+1]]<=t<knt[[i+2]], 1, 0];
n=4; k=4; (* Note: 0<=k<=n *)
knt={-3,-2,-1,0,1,2,3,4,5,6,7,8}; (* knots for nonuniform case *)
bspl[i,k,t] (* assign a value to i *)
```

Figure 7.20, page 292, Eight-Point Nonuniform B-Spline Example; Code for Blending Functions.

```
(* Rational B-spline example. w_2=0, .5, 1, 5 (Slow!) *)
Clear[bspl,knt,w,pnts,cur1,cur2,cur3,cur4,R] (* weight functions *)
bspl[i_,k_,t_]:=If[knt[[i+k]]==knt[[i+1]],0, (* 0<=i<=n *)
 bspl[i,k-1,t] (t-knt[[i+1]])/(knt[[i+k]]-knt[[i+1]])] \
 +If[knt[[i+1+k]]==knt[[i+2]],0,
 bspl[i+1,k-1,t] (knt[[i+1+k]]-t)/(knt[[i+1+k]]-knt[[i+2]])];
bspl[i_,1,t_]:=If[knt[[i+1]]<=t<knt[[i+2]], 1, 0];
R[i_,t_]:=(w[[i+1]] bspl[i,k,t])/Sum[w[[j+1]] bspl[j,k,t], {j,0,n}];
n=4; k=3; w={1,1,0,1,1}; (* weights *)
knt={0,0,0,1,2,3,3,3}; (* knots *)
pnts={{0,0}, {0,1}, {1,0}, {2,1}, {2,0}};
cur1=ParametricPlot[Sum[(R[i,t] pnts[[i+1]]), {i,0,n}], {t,0,3},
 PlotRange->All, DisplayFunction->Identity, Compiled->False];
w[[3]]=0.5;
cur2=ParametricPlot[Sum[(R[i,t] pnts[[i+1]]), {i,0,n}], {t,0,3},
 PlotRange->All, DisplayFunction->Identity, Compiled->False];
w[[3]]=1;
cur3=ParametricPlot[Sum[(R[i,t] pnts[[i+1]]), {i,0,n}], {t,0,3},
 PlotRange->All, DisplayFunction->Identity, Compiled->False];
w[[3]]=5;
cur4=ParametricPlot[Sum[(R[i,t] pnts[[i+1]]), {i,0,n}], {t,0,3},
 PlotRange->All, DisplayFunction->Identity, Compiled->False];
Show[cur1,cur2,cur3,cur4, PlotRange->All, DefaultFont->{"cmr10", 10},
 DisplayFunction->$DisplayFunction];
```

Figure 7.21, page 306, Effects of Varying Weight w_2.

```
(* One third of a circle done by rational B-spline *)
P0={0,-1}; P1={-1.732,-1}; P2={-0.866,0.5}; w1=0.5;
```

```
pnts=ListPlot[{P0,P1,P2}, Prolog->PointSize[.04], DisplayFunction->Identity];
axs={AbsoluteThickness[1], Line[{P0,P1,P2}]};
th=ParametricPlot[((1-t)^2 P0+2w1 t(1-t)P1+t^2 P2)/((1-t)^2+2w1 t(1-t)+t^2),
 {t,0,1}, PlotRange->All, DisplayFunction->Identity, Compiled->False];
Show[Graphics[axs],th,pnts, PlotRange->All, DisplayFunction->$DisplayFunction];
```

Figure 7.23, page 308, Control Points for Circles.

```
(* BiQuadratic B-spline Patch Example *)
<<:Graphics:ParametricPlot3D.m
Clear[T,Pnts,Q,comb,g1,g2];
T[t_]:={t^2,t,1};
Pnts={{{0,0,0},{0,1.5,0},{0,2,0}},{{1,0,0},{1,1,1},{1,2,0}},
 {{2,0,0},{2,0.5,0},{2,2,0}}};
Q={{1,-2,1},{-2,2,0},{1,1,0}};
g1=Graphics3D[{AbsolutePointSize[3], Table[Point[Pnts[[i,j]]],{i,1,3},{j,1,3}]}];
comb[i_]:=((1/4)T[u].Q.Pnts)[[i]] (Transpose[Q].T[w])[[i]]
g2=ParametricPlot3D[comb[1]+comb[2]+comb[3], {u,0,1},{w,0,1}, AspectRatio->Automatic,
 Ticks->{{0,1,2},{0,1,2},{0,1}}, Compiled->False, DisplayFunction->Identity];
Show[g2,g1, DisplayFunction->$DisplayFunction, ViewPoint->{-0.196, -4.177, 1.160},
 PlotRange->All, DefaultFont->{"cmr10", 10}];
```

Figure 7.25, page 310, A Biquadratic B-Spline Surface Patch.

```
0 0 0 0 1 1 0 2 1 0 2 0
1 0 0 1 1 2 1 2 1 1 3 2
2 0 0 2 1 3 2 2 2 2 3 3
3 0 0 3 1 2 3 2 1 3 3 2
4 0 0 4 1 1 4 2 1 4 2 0
```

```
(* a general uniform B-spline surface patch *)
Clear[bsplSurf,surpnts,bspl,g1,g2,knt,i,j,km,kn,m,n,u,w]
bspl[i_,k_,t_]:=bspl[i,k-1,t] (t-knt[[i+1]])/(knt[[i+k]]-knt[[i+1]]) \
 +bspl[i+1,k-1,t] (knt[[i+1+k]]-t)/(knt[[i+1+k]]-knt[[i+2]]) (* 0<=i<=n *)
bspl[i_,1,t_]:=If[knt[[i+1]]<=t<knt[[i+2]], 1, 0];
n=3; kn=3; m=4; km=3; (* Note: 0<=kn<=n 0<=km<=m *)
knt=Table[i, {i,0,m+km}]; (* uniform knots *)
(* Input triplets from data file *)
surpnts=ReadList["surf.pnts", {Number,Number,Number}, RecordLists->True];
bsplSurf[u_,w_]:=Sum[Sum[surpnts[[i+1,j+1]]bspl[i,km,u],{i,0,m}]bspl[j,kn,w],{j,0,n}]
g1=Graphics3D[{AbsolutePointSize[3], Table[Point[surpnts[[i,j]]],{i,1,5},{j,1,4}]}];
g2=ParametricPlot3D[bsplSurf[u,w], {u,km-1,m+1},{w,kn-1,n+1},
 DisplayFunction->Identity,
 AspectRatio->Automatic, Compiled->False];
Show[g1,g2, PlotRange->All, DisplayFunction->$DisplayFunction,
 DefaultFont->{"cmr10", 10}, ViewPoint->{1.389, -3.977, 1.042}];
```

Figure 7.27, page 316, A Quadratic-Cubic B-Spline Surface Patch.

Chapter 8

```
(* Chaikin algorithm for a control polygon *)
n=4;
(* p={p0,p1,p2,p3,p4,p5}; *)
p={{0,0},{0,4},{3,4},{4,0},{6,6}};
Show[Graphics[Line[p]]]
q=Table[If[OddQ[i], (* then *){(3p[[i]]+p[[i+1]])/4, (p[[i]]+3p[[i+1]])/4},
```

```
(* else *) {(3p[[i]]+p[[i+1]])/4, (p[[i]]+3p[[i+1]])/4}], {i,1,n}];
q=Flatten[q,1]
Show[Graphics[{AbsoluteDashing[{2,2}], Line[p]}], Graphics[Line[q]]];
r=Table[If[OddQ[i], (* then *){(3q[[i]]+q[[i+1]])/4, (q[[i]]+3q[[i+1]])/4},
 (* else *) {(3q[[i]]+q[[i+1]])/4, (q[[i]]+3q[[i+1]])/4}], {i,1,2n-1}];
r=Flatten[r,1]
Show[Graphics[{AbsoluteDashing[{2,2}], Line[p]}], Graphics[Line[r]]];
```

Figure 8.5, page 323, Chaikin's Algorithm for a Control Polygon.

```
(* reparametrize biquadratic B-spline surface *)
Clear[a,b,c,d,A,B,TB,H,M,P,Q];
M={{1,-2,1},{-2,2,0},{1,1,0}}/2;
A={{(b-a)^2,0,0},{2a(b-a),b-a,0},{a^2,a,1}};
(* B=MatrixForm[Simplify[Inverse[M].A.M]] *)
B={{((1 - a)*(1 - 2*a + b))/2, (1 + 3*a - 4*a^2 - b + 2*a*b)/2,
  a^2 - (a*b)/2}, {1/2 - a/2 - b/2 + (a*b)/2, (1 + a + b - 2*a*b)/2,
  (a*b)/2}, {((1 + a - 2*b)*(1 - b))/2, (1 - a + 3*b + 2*a*b - 4*b^2)/2,
  -(a*b)/2 + b^2}};
TB={{((1 - c)*(1 - 2*c + d))/2, 1/2 - c/2 - d/2 + (c*d)/2,
  ((1 + c - 2*d)*(1 - d))/2},
 {(1 + 3*c - 4*c^2 - d + 2*c*d)/2, (1 + c + d - 2*c*d)/2,
  (1 - c + 3*d + 2*c*d - 4*d^2)/2},
 {c^2 - (c*d)/2, (c*d)/2, -(c*d)/2 + d^2}};
P={{P00,P01,P02},{P10,P11,P12},{P20,P21,P22}};
Q=Simplify[B.P.TB]
a=0; b=.5; c=0; d=.5; Q
```

Figure 8.9, page 333, Code for the Nine Control Points of the "Upper-Left" Patch.

```
(* reparametrize bicubic B-spline surface *)
Clear[a,b,c,d,A,B,TB,H,M,P,Q];
M={{-1,3,-3,1},{3,-6,3,0},{-3,0,3,0},{1,4,1,0}}/6;
A={{(b-a)^3,0,0,0},{3a(b-a)^2,(b-a)^2,0,0},{3a^2(b-a),2a(b-a),b-a,0},{a^3,a^2,a,1}};
(*B=Simplify[Inverse[M].A.M] *)
B={{((1 - a)*(1 - 5*a + 6*a^2 + 3*b - 7*a*b + 2*b^2))/6,
  (4 - 22*a^2 + 18*a^3 + 20*a*b - 21*a^2*b - 4*b^2 + 6*a*b^2)/6,
  1/6 + a + (11*a^2)/6 - 3*a^3 - b/2 - (5*a*b)/3 + (7*a^2*b)/2 + b^2/3 -
   a*b^2, a^3 - (7*a^2*b)/6 + (a*b^2)/3},
 {((-1 + a)*(-1 + 2*a - 2*a*b + b^2))/6,
  (4 - 4*a^2 - 4*a*b + 6*a^2*b + 2*b^2 - 3*a*b^2)/6,
  1/6 + a/2 + a^2/3 + (a*b)/3 - a^2*b - b^2/6 + (a*b^2)/2,
  (a*(2*a - b)*b)/6}, {((-1 + a)*(1 + a - 2*b)*(-1 + b))/6,
  (4 + 2*a^2 - 4*a*b - 3*a^2*b - 4*b^2 + 6*a*b^2)/6,
  1/6 - a^2/6 + b/2 + (a*b)/3 + (a^2*b)/2 + b^2/3 - a*b^2,
  (a*b*(-a + 2*b))/6}, {((1 - b)*(1 + 3*a + 2*a^2 - 5*b - 7*a*b + 6*b^2))/
   6, (4 - 4*a^2 + 20*a*b + 6*a^2*b - 22*b^2 - 21*a*b^2 + 18*b^3)/6,
  1/6 - a/2 + a^2/3 + b - (5*a*b)/3 - a^2*b + (11*b^2)/6 + (7*a*b^2)/2 -
   3*b^3, (a^2*b)/3 - (7*a*b^2)/6 + b^3}};
TB={{((1 - a)*(1 - 5*a + 6*a^2 + 3*b - 7*a*b + 2*b^2))/6,
  ((-1 + a)*(-1 + 2*a - 2*a*b + b^2))/6,
  ((-1 + a)*(1 + a - 2*b)*(-1 + b))/6,
  ((1 - b)*(1 + 3*a + 2*a^2 - 5*b - 7*a*b + 6*b^2))/6},
 {(4 - 22*a^2 + 18*a^3 + 20*a*b - 21*a^2*b - 4*b^2 + 6*a*b^2)/6,
  (4 - 4*a^2 - 4*a*b + 6*a^2*b + 2*b^2 - 3*a*b^2)/6,
  (4 + 2*a^2 - 4*a*b - 3*a^2*b - 4*b^2 + 6*a*b^2)/6,
  (4 - 4*a^2 + 20*a*b + 6*a^2*b - 22*b^2 - 21*a*b^2 + 18*b^3)/6},
 {1/6 + a + (11*a^2)/6 - 3*a^3 - b/2 - (5*a*b)/3 + (7*a^2*b)/2 +
   b^2/3 - a*b^2, 1/6 + a/2 + a^2/3 + (a*b)/3 - a^2*b - b^2/6 +
   (a*b^2)/2, 1/6 - a^2/6 + b/2 + (a*b)/3 + (a^2*b)/2 + b^2/3 - a*b^2,
  1/6 - a/2 + a^2/3 + b - (5*a*b)/3 - a^2*b + (11*b^2)/6 + (7*a*b^2)/2 -
```

```
  3*b^3}, {a^3 - (7*a^2*b)/6 + (a*b^2)/3, (a*(2*a - b)*b)/6,
  (a*b*(-a + 2*b))/6, (a^2*b)/3 - (7*a*b^2)/6 + b^3}};
P={{P30,P31,P32,P33},{P20,P21,P22,P23},{P10,P11,P12,P13},{P00,P01,P02,P03}};
Q=Simplify[B.P.TB]
a=0; b=.5; c=0; d=.5; Q
```

Figure 8.13, page 339, Code for the 16 Control Points of the “Uper-Left” Patch.

Chapter 9

```
(* 2 sweep surface examples *)
alf=1;
ParametricPlot3D[
{u Cos[2Pi w],u Sin[2Pi w], alf w},{u,0,1},{w,0,1}, DefaultFont->{"cmr10", 10},
 Compiled->False, ViewPoint->{3.369, -2.693, 0.479}, PlotPoints->20]
 m={-3u^3+6u^2+3u,-3u^3+3u^2+1,3u^2-3u+1,1}.
 {{1,0,0,0},{0,1,0,0},{0,0,1,0},{-4w^3+3w^2+3w,-6w^2+6w,-2w^3+3w,1}};
 ParametricPlot3D[Drop[m,-1],{u,0,1},{w,0,1}, DefaultFont->{"cmr10", 10},
 Compiled->False, ViewPoint->{4.068, -1.506, 0.133}, PlotPoints->20]
```

Figure 9.1, page 349, Two Sweep Surfaces.

```
(* Mobius strip as a sweep surface *)
<<:Graphics:ParametricPlot3D.m
Clear[r,roty,rotz,segm];
segm[t_]:={t,0,0}; (* a short line segment *)
roty[phi_]:={{Cos[phi],0,-Sin[phi]},{0,1,0},{Sin[phi],0,Cos[phi]}};
rotz[phi_]:={{Cos[phi],-Sin[phi],0},{Sin[phi],Cos[phi],0},{0,0,1}};
ParametricPlot3D[Evaluate[rotz[phi].(roty[phi/2].segm[t]+{20,0,0})],
 {phi,0,2Pi}, {t,-3,3}, Boxed->True, PlotPoints->{35,2}, Axes->False]
Show[{%,Graphics3D[{AbsoluteThickness[1], (* show the 3 axes *)
 Line[{{0,0,30},{0,0,0},{30,0,0},{0,0,0},{0,30,0}}]}]}]
```

Figure 9.2, page 350, A Möbius Strip.

```
(* Sweep surface example. Lofted surface with scaling transform *)
<<:Graphics:ParametricPlot3D.m
pnts={{-1,-1,0},{1,-1,0},{-1,1,0},{0,1,1},{1,1,0}};
{2u-1,2w-1,4u w(1-u)}.{{w,0,0},{0,1,0},{0,0,1}};
g1=ParametricPlot3D[%, {u,0,1},{w,0,1}, Compiled->False,
DefaultFont->{"cmr10", 10},
 AspectRatio->Automatic, Ticks->{{0,1},{0,1},{0,1}}]
g2=Graphics3D[{AbsolutePointSize[4], Table[Point[pnts[[i]]],{i,1,5}]}]
Show[g1,g2, ViewPoint->{-0.139, -1.179, 1.475}]
```

Figure 9.3, page 351, A Lofted Swept Surface.

```
(* A Sweep Surface.
 Curve Cu[u,w] times matrix Trn[w] *)
<<:Graphics:ParametricPlot3D.m;
Clear[Cu,Trn];
Cu[u_,w_]:={u,1,u+2}w+{-u,1,u-2}(1-w);
Trn[w_]:={
 {Cos[2Pi w],Sin[2Pi w],0},
```

```
 {-Sin[2Pi w],Cos[2Pi w],0},
 {0,0,1}};
ParametricPlot3D[
 {Cu[u,w].Trn[w][[1]],Cu[u,w].Trn[w][[2]],
 Cu[u,w].Trn[w][[3]]},
{u,0,1,.2},{w,0,1,.2}, Ticks->None,
 PlotRange->All, AspectRatio->Automatic,
 RenderAll->False, Prolog->AbsoluteThickness[.4],
 ViewPoint->{-0.510, -1.365, 1.210}]
```

Figure 9.4, page 352, Sweeping while Rotating.

```
(* A Chalice *)
<<:Graphics:SurfaceOfRevolution.m
(* the profile *)
ParametricPlot[{.5u^3-.3u^2-.5u-.2,u+1},{u,-1,1},
 AspectRatio->Automatic]
(* the surface *)
SurfaceOfRevolution[{.5u^3-.3u^2-.5u-.2,u+1},{u,-1,1}, PlotPoints->40]
```

Figure 9.8, page 356, A Chalice as a Surface of Revolution.

```
<<:Graphics:ParametricPlot3D.m;   (* Surface of revolution *)
Clear[basis,Cubi]; (* as a combination of 2 cubic B-splines *)
(* matrix 'basis' has dimensions 4x4x3 *)
basis={{{0,0,0},{0,-3/2,0},{0,-3/2,3},{0,0,3}}
 ,{{0,0,0},{-3/2,0,0},{-3/2,0,3},{0,0,3}}
 ,{{0,0,0},{0,3/2,0},{0,3/2,3},{0,0,3}},{{0,0,0}
 ,{3/2,0,0},{3/2,0,3},{0,0,3}}};
Cubi={{-1,3,-3,1},{3,-6,3,0},{-3,0,3,0},{1,4,1,0}};
prt[i_]:=basis[[Range[1,4],Range[1,4],i]];
(* 'prt' extracts component i from the 3rd dimen of 'basis' *)
coord[i_]:={u^3,u^2,u,1}.Cubi.prt[i].Transpose[Cubi].{w^3,w^2,w,1};
ParametricPlot3D[{coord[1],coord[2],coord[3]}/36,
{u,0,1,.1},{w,0,1,.1},
Prolog->AbsoluteThickness[.5],ViewPoint->{1.736, -0.751, -0.089}]
```

Figure 9.10, page 360, A Quarter-Circle Surface of Revolution made of B-Splines.

Answers to Exercises

```
(* A lofted surface example. Bottom boundary curve is straight *)
pnts={{-1,-1,0},{1,-1,0},{-1,1,0},{0,1,1},{1,1,0}};
g1=Graphics3D[{AbsolutePointSize[5],
         Table[Point[pnts[[i]]],{i,1,5}]}]
g2=ParametricPlot3D[{2u-1,2w-1,4u w(1-u)}, {u,0,1},{w,0,1},
DefaultFont->{"cmr10", 10}, DisplayFunction->Identity,
 AspectRatio->Automatic, Ticks->{{0,1},{0,1},{0,1}}]
Show[g1,g2, ViewPoint->{-0.139, -1.179, 1.475}]
```

Figure Ans.3, page 400: A Lofted Surface.

```
Clear[Nh,p,pnts,U,W];
```

```
p00={0,0,0}; p10={1,0,1}; p20={2,0,1}; p30={3,0,0};
p01={0,1,1}; p11={1,1,2}; p21={2,1,2}; p31={3,1,1};
p02={0,2,1}; p12={1,2,2}; p22={2,2,2}; p32={3,2,1};
p03={0,3,0}; p13={1,3,1}; p23={2,3,1}; p33={3,3,0};
Nh={{-4.5,13.5,-13.5,4.5},{9,-22.5,18,-4.5},
 {-5.5,9,-4.5,1},{1,0,0,0}};
pnts={{p33,p32,p31,p30},{p23,p22,p21,p20},
 {p13,p12,p11,p10},{p03,p02,p01,p00}};
U[u_]:={u^3,u^2,u,1};   W[w_]:={w^3,w^2,w,1};
(* prt [i] extracts component i from the 3rd dimen of P *)
prt[i_]:=pnts[[Range[1,4],Range[1,4],i]];
p[u_,w_]:={U[u].Nh.prt[1].Transpose[Nh].W[w],
 U[u].Nh.prt[2].Transpose[Nh].W[w], \
 U[u].Nh.prt[3].Transpose[Nh].W[w]};
 g1=ParametricPlot3D[p[u,w], {u,0,1},{w,0,1},
 Compiled->False, DisplayFunction->Identity];
g2=Graphics3D[{AbsolutePointSize[2],
 Table[Point[pnts[[i,j]]],{i,1,4},{j,1,4}]}];
Show[g1,g2, ViewPoint->{-2.576, -1.365, 1.718}]
```

Figure Ans.4, page 407, An Interpolating Bicubic Surface Patch and Code.

```
(* Heart-shaped Bezier curve *)
n=9; ppr=130;
pnts={{0,0},{-ppr,70},{-ppr,200},{0,200},{250,0},{-250,0},{0,200},
 {ppr,200},{ppr,70},{0,0}};
pwr[x_,y_]:=If[x==0 && y==0, 1, x^y];
bern[n_,i_,t_]:=Binomial[n,i]pwr[t,i]pwr[1-t,n-i]
bzCurve[t_]:=Sum[pnts[[i+1]]bern[n,i,t], {i,0,n}]
g1=ListPlot[pnts, Prolog->AbsolutePointSize[4], PlotRange->All,
 AspectRatio->Automatic, DisplayFunction->Identity]
g2=ParametricPlot[bzCurve[t], {t,0,1}, Compiled->False,
 PlotRange->All, AspectRatio->Automatic, DisplayFunction->Identity]
g3=Graphics[{AbsoluteDashing[{1,2,5,2}], Line[pnts]}]
Show[g1,g2,g3, DisplayFunction->$DisplayFunction,
 DefaultFont->{"cmr10", 10}]
```

Figure Ans.7, page 417, A Heart-Shaped Bézier Curve.

```
q0={0,0}; q1={1,1}; q2={2,1}; q3={3,0};
p0=q0; p1={1,3/2}; p2={2,3/2}; p3=q3;
c[t_]:=(1-t)^3 p0+3t(1-t)^2 p1+3t^2(1-t) p2+t^3 p3
g1=ListPlot[{p0,p1,p2,p3,q1,q2},
 Prolog->AbsolutePointSize[4], PlotRange->All,
 AspectRatio->Automatic, DisplayFunction->Identity]
g2=ParametricPlot[c[t], {t,0,1}, DisplayFunction->Identity]
Show[g1,g2, DisplayFunction->$DisplayFunction]
```

Figure Ans.11, page 426, An Interpolating Bézier Curve.

```
(* Effects of varying weights in Rational Cubic Bezier curve *)
```

```
Clear[RatCurve,g1,g2,w];
pnts={{0,0},{.2,1},{.8,1},{1,0}};
w={1,1,1,1}; (* Four weights for a cubic curve *)
pwr[x_,y_]:=If[x==0 && y==0, 1, x^y];
bern[n_,i_,t_]:=Binomial[n,i]pwr[t,i]pwr[1-t,n-i] (* t^i*(1-t)^(n-i) *)
RatCurve[t_]:=Sum[(w[[i+1]]pnts[[i+1]]bern[3,i,t])/(Sum[w[[j+1]]bern[3,j,t],
 {j,0,3}]), {i,0,3}];
g1=ListPlot[pnts, Prolog->AbsolutePointSize[4], PlotRange->All,
 AspectRatio->Automatic, DisplayFunction->Identity]
g2=ParametricPlot[RatCurve[t], {t,0,1}, Compiled->False,
 PlotRange->All, AspectRatio->Automatic, DisplayFunction->Identity]
w={1,2,1,1}; (* change weights *)
g3=ParametricPlot[RatCurve[t], {t,0,1}, Compiled->False,
 PlotRange->All, AspectRatio->Automatic, DisplayFunction->Identity]
w={1,3,1,1}; (* increase w1 *)
g4=ParametricPlot[RatCurve[t], {t,0,1}, Compiled->False,
 PlotRange->All, AspectRatio->Automatic, DisplayFunction->Identity]
w={1,4,1,1}; (* increase w1 *)
g5=ParametricPlot[RatCurve[t], {t,0,1}, Compiled->False,
 PlotRange->All, AspectRatio->Automatic, DisplayFunction->Identity]
Show[g1,g2,g3,g4,g5, DisplayFunction->$DisplayFunction,
 DefaultFont->{"cmr10",10}]

(* Effects of moving a control point in Rational Cubic Bezier curve *)
Clear[RatCurve,g1,g2,w];
pnts={{0,0},{.2,.8},{.8,.8},{1,0}};
w={1,1,1,1}; (* Four weights for a cubic curve *)
pwr[x_,y_]:=If[x==0 && y==0, 1, x^y];
bern[n_,i_,t_]:=Binomial[n,i]pwr[t,i]pwr[1-t,n-i] (* t^i*(1-t)^(n-i) *)
RatCurve[t_]:=Sum[(w[[i+1]]pnts[[i+1]]bern[3,i,t])/(Sum[w[[j+1]]bern[3,j,t],
 {j,0,3}]), {i,0,3}];
g1=ListPlot[pnts, Prolog->AbsolutePointSize[4], PlotRange->All,
 AspectRatio->Automatic, DisplayFunction->Identity]
g2=ParametricPlot[RatCurve[t], {t,0,1}, Compiled->False,
 PlotRange->All, AspectRatio->Automatic, DisplayFunction->Identity]
pnts={{0,0},{.2,.8},{.86,.86},{1,0}};
g3=ParametricPlot[RatCurve[t], {t,0,1}, Compiled->False,
 PlotRange->All, AspectRatio->Automatic, DisplayFunction->Identity]
pnts={{0,0},{.2,.8},{.93,.93},{1,0}};
g4=ParametricPlot[RatCurve[t], {t,0,1}, Compiled->False,
 PlotRange->All, AspectRatio->Automatic, DisplayFunction->Identity]
pnts={{0,0},{.2,.8},{1,1},{1,0}};
g5=ParametricPlot[RatCurve[t], {t,0,1}, Compiled->False,
 PlotRange->All, AspectRatio->Automatic, DisplayFunction->Identity]
Show[g1,g2,g3,g4,g5, DisplayFunction->$DisplayFunction,
 DefaultFont->{"cmr10",10}]
```

Figure Ans.12, page 427, Code for Figure 6.19.

```
(* A Rational closed Bezier Surface *)
Clear[pwr,bern,spnts,n,m,wt,bzSurf,cpnts,patch,vlines,hlines,axes];
<<:Graphics:ParametricPlot3D.m
r=1; h=3; (* radius & height of cylinder *)
spnts={{{r,0,0},{0,2r,0},{-r,0,0},{0,-2r,0},{r,0,0}},
 {{r,0,h},{0,2r,h},{-r,0,h},{0,-2r,h},{r,0,h}}};
```

```
m=Length[spnts[[1]]]-1; n=Length[Transpose[spnts][[1]]]-1;
wt=Table[1, {i,1,n+1},{j,1,m+1}];
pwr[x_,y_]:=If[x==0 && y==0, 1, x^y];
bern[n_,i_,u_]:=Binomial[n,i]pwr[u,i]pwr[1-u,n-i]
bzSurf[u_,w_]:=
 Sum[wt[[i+1,j+1]]spnts[[i+1,j+1]]bern[n,i,u]bern[m,j,w], {i,0,n}, {j,0,m}]/
 Sum[wt[[i+1,j+1]]bern[n,i,u]bern[m,j,w], {i,0,n}, {j,0,m}];
patch=ParametricPlot3D[bzSurf[u,w],{u,0,1}, {w,0,1},
 Compiled->False, DisplayFunction->Identity];
cpnts=Graphics3D[{AbsolutePointSize[4], (* control points *)
 Table[Point[spnts[[i,j]]], {i,1,n+1},{j,1,m+1}]}];
vlines=Graphics3D[{AbsoluteThickness[1], (* control polygon *)
 Table[Line[{spnts[[i,j]],spnts[[i+1,j]]}], {i,1,n}, {j,1,m+1}]}];
hlines=Graphics3D[{AbsoluteThickness[1],
 Table[Line[{spnts[[i,j]],spnts[[i,j+1]]}], {i,1,n+1}, {j,1,m}]}];
maxx=Max[Flatten[Table[Part[spnts[[i,j]], 1], {i,1,n+1}, {j,1,m+1}]]];
maxy=Max[Flatten[Table[Part[spnts[[i,j]], 2], {i,1,n+1}, {j,1,m+1}]]];
maxz=Max[Flatten[Table[Part[spnts[[i,j]], 3], {i,1,n+1}, {j,1,m+1}]]];
axes=Graphics3D[{AbsoluteThickness[1.5], (* the coordinate axes *)
 Line[{{0,0,maxz},{0,0,0},{maxx,0,0},{0,0,0},{0,maxy,0}}]}];
Show[cpnts,hlines,vlines,axes,patch, PlotRange->All,DefaultFont->{"cmr10", 10},
 DisplayFunction->$DisplayFunction, ViewPoint->{0.998, 0.160, 4.575},Shading->False];
```

Figure Ans.13, page 428, A Closed Rational Bézier Surface Patch.

```
P0300={3,3,0};
P0210={2,2,0}; P1200={4,2,1};
P0120={1,1,0}; P1110={3,1,1}; P2100={5,1,2};
P0030={0,0,0}; P1020={2,0,1}; P2010={4,0,2}; P3000={6,0,3};
n=3; u=1/6; v=2/6; w=3/6;
P0021=u P1020+v P0120+w P0030;
P1011=u P2010+v P1110+w P1020;
P2001=u P3000+v P2100+w P2010;
P0111=u P1110+v P0210+w P0120;
P1101=u P2100+v P1200+w P1110;
P0201=u P1200+v P0300+w P0210;
P0012=u P1011+v P0111+w P0021;
P1002=u P2001+v P1101+w P1011;
P0102=u P1101+v P0201+w P0111;
P0003=u P1002+v P0102+w P0012
B[i_,j_,k_]:=(n!/(i! j! k!))u^i v^j w^k;
P0030 B[0,0,3]+P1020 B[1,0,2]+P2010 B[2,0,1]+P3000 B[3,0,0]+
P0120 B[0,1,2]+P1110 B[1,1,1]+P2100 B[2,1,0]+
P0210 B[0,2,1]+P1200 B[1,2,0]+P0300 B[0,3,0]
```

Figure Ans.15, page 430, Triangular Bézier Patch Subdivision Exercise.

```
(* Exercise. 8 points, 5-segment uniform B-spline curve, compared to the Bezier
 curve for the same 8 points *)
Clear[p1,p2,p3,p4,p5,bez,l1,g1,g2,g3,g4,g5,g6];
pnts={{1,0},{2,1},{4,0},{4,1}};
p1[t_]:={t^3+6,t^3}/6;
p2[t_]:={3t^2+3t+7,-3t^3+3t^2+3t+1}/6;
p3[t_]:={-3t^3+3t^2+9t+13,4t^3-6t^2+4}/6;
p4[t_]:={2t^3-6t^2+6t+22,-3t^3+6t^2+2}/6;
p5[t_]:={24,t^3-3t^2+3t+5}/6;
bez[t_]:={-3t^3+3t^2+3t+1,4t^3-6t^2+3t};
l1=ListPlot[pnts, Prolog->PointSize[.01], DisplayFunction->Identity];
g1=ParametricPlot[p1[t], {t,0,.97}, Compiled->False, DisplayFunction->Identity];
g2=ParametricPlot[p2[t], {t,0,.97}, Compiled->False, DisplayFunction->Identity];
```

```
g3=ParametricPlot[p3[t], {t,0,.97}, Compiled->False, DisplayFunction->Identity];
g4=ParametricPlot[p4[t], {t,0,.97}, Compiled->False, DisplayFunction->Identity];
g5=ParametricPlot[p5[t], {t,0,.97}, Compiled->False, DisplayFunction->Identity];
g6=ParametricPlot[bez[t], {t,0,1}, PlotStyle->AbsoluteDashing[{2,2}],
 Compiled->False, DisplayFunction->Identity];
 (* Now the degree-7 Bezier curve *)
pnts={{1,0},{1,0},{1,0},{2,1},{4,0},{4,1},{4,1},{4,1}};
pwr[x_,y_]:=If[x==0 && y==0, 1, x^y];
bern[n_,i_,t_]:=Binomial[n,i]pwr[t,i]pwr[1-t,n-i] (* t^i x (1-t)^(n-i) *)
bzCurve[t_]:=Sum[pnts[[i+1]]bern[7,i,t], {i,0,7}]
g7=ParametricPlot[bzCurve[t], {t,0,1}, Compiled->False,
 PlotStyle->AbsoluteDashing[{1,2,2,2}], PlotRange->All,
 AspectRatio->Automatic, DisplayFunction->Identity];
Show[l1,g1,g2,g3,g4,g5,g6,g7, PlotRange->All, DisplayFunction->$DisplayFunction,
AspectRatio->Automatic, DefaultFont->{"cmr10", 10}];
```

Figure Ans.18, page 434, Comparing a Uniform B-spline and a Bézier Curve for Eight Points.

```
a={{4,4,0},{1,6,1},{0,4,4}}/8; {p10,p11,p12}=a.{p00,p01,p02};
{p12,p13,p14}=a.{p01,p02,p03}; {p20,p21,p22}=a.{p10,p11,p12};
{p22,p23,p24}=a.{p11,p12,p13}; {p24,p25,p26}=a.{p12,p13,p14};
{p30,p31,p32}=a.{p20,p21,p22}; {p32,p33,p34}=a.{p21,p22,p23};
{p34,p35,p36}=a.{p22,p23,p24}; {p36,p37,p38}=a.{p23,p24,p25};
{p38,p39,p310}=a.{p24,p25,p26}; Simplify[(p36+4 p37+p38)/6]
```

Figure Ans.19, page 439, Code for Exercise 8.6

```
<<:Graphics:ParametricPlot3D.m;
ParametricPlot3D[{3u,Sin[w],w}, {u,0,1},{w,0,4Pi},
Ticks->False, AspectRatio->Automatic]
```

Figure Ans.20, page 440, A Sweep Surface.

```
R=10; r=2;  (* The Torus as a surface of revolution *)
ParametricPlot3D[
 {(R+r Cos[2Pi u])Cos[2Pi w],-(R+r Cos[2Pi u])Sin[2Pi w],
 r Sin[2Pi u]},{u,0,1},{w,0,1},
 ViewPoint->{-0.028, -4.034, 1.599}]
```

Figure Ans.21, page 441, The Torus as a Surface of Revolution.

End of Listing