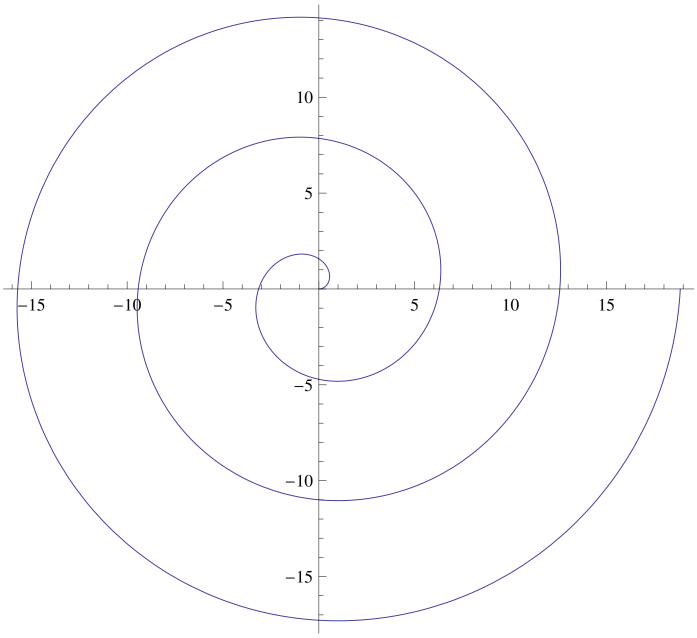
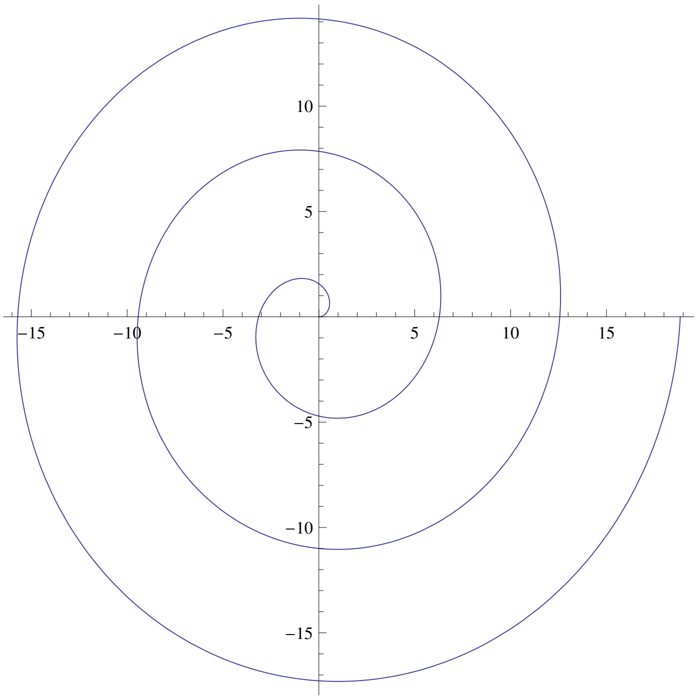
```
(* WQ, FS 2013,
Verwendung von AspectRatio zur Scalierung von Abbildungen  *)

ParametricPlot[{t Cos[t], t Sin[t]}, {t, 0, 6 Pi}]
```

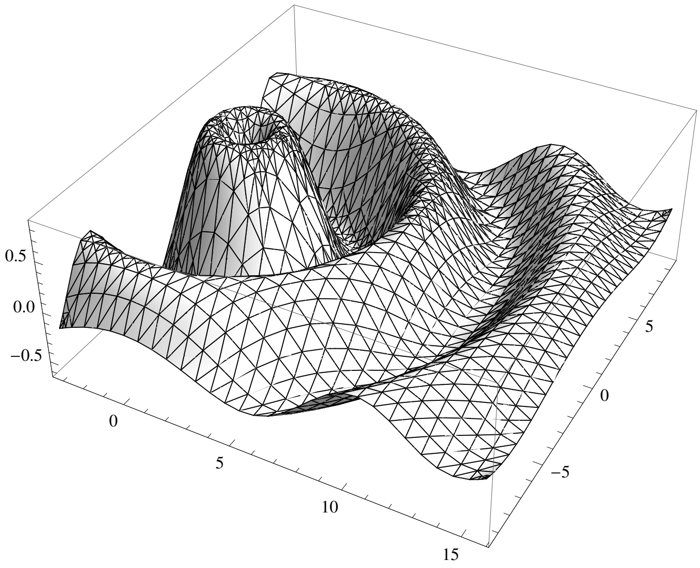


```
AbsoluteOptions[%, AspectRatio]

{AspectRatio → 0.91011}

ParametricPlot[{t Cos[t], t Sin[t]}, {t, 0, 6 Pi}, AspectRatio → 1]
```

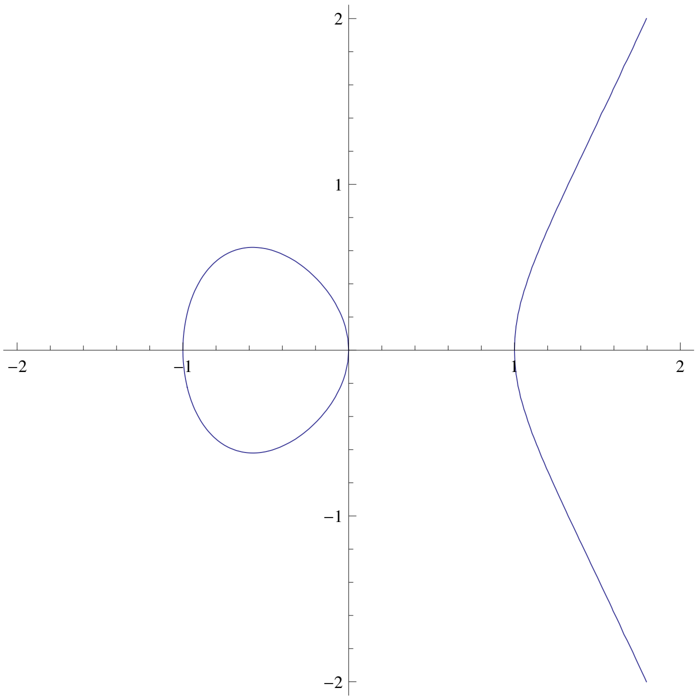


```
(* Anzeigen der Triangulation - die Mma verwendet - mit Mesh *)
```

```
Plot3D[Sin[Sqrt[x^2 + y^2]] * Exp[-0.1 * Sqrt[x^2 + y^2]], {x, -Pi, 5 Pi},
 {y, -3 Pi, 3 Pi}, PlotPoints → 30, Mesh → All, Lighting → "Neutral"]
```

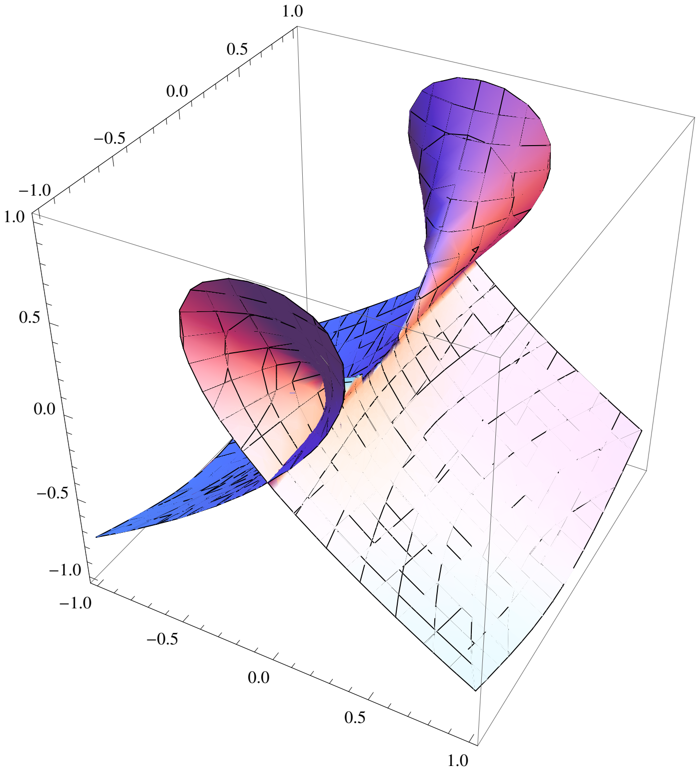


```
(* Elliptische Kurve mit CountourPlot *)

ContourPlot[y^2 == x^3 - x, {x, -2, 2}, {y, -2, 2}, Frame → False, Axes → True]
```

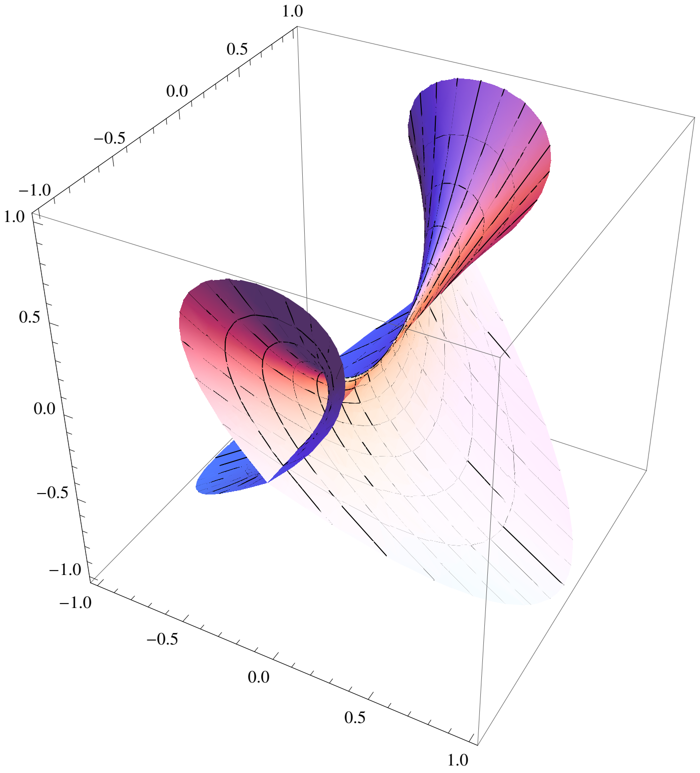


```
(*  verschiedene Versionen einer Flaeche mit Name Schweinsohr *)
```

```
ContourPlot3D[x^2 - y^2 z^2 + z^3 == 0, {x, -1, 1}, {y, -1, 1}, {z, -1, 1}]
```

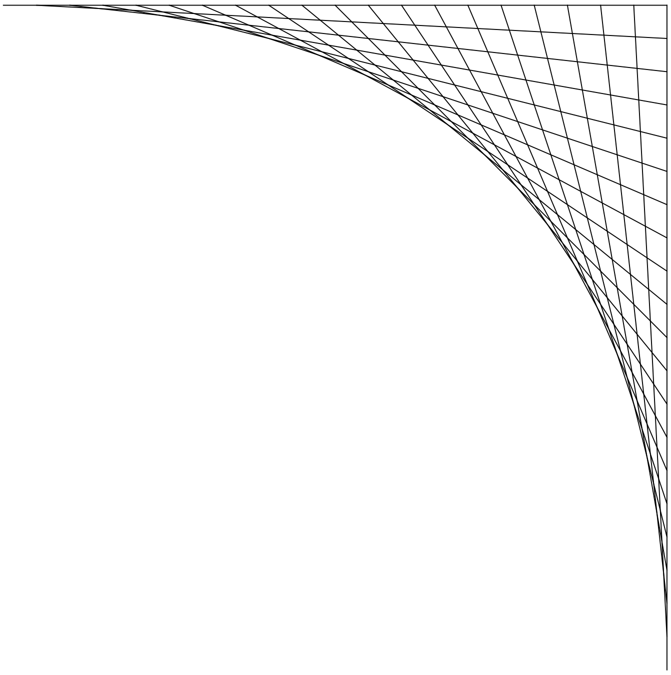


```
ParametricPlot3D[{t (u^2 - t^2), u, u^2 - t^2}, {t, -1, 1}, {u, -1, 1}]
```



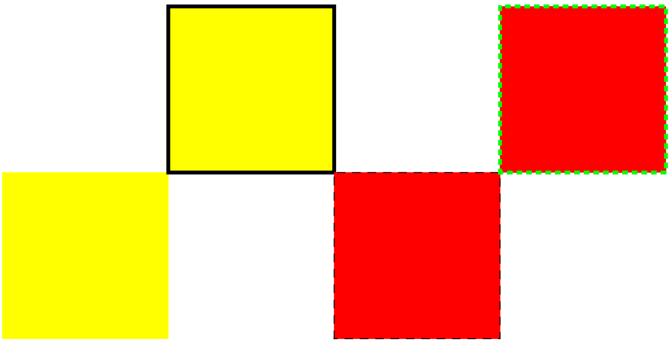
```
(* Verwendung von GraphicsPrimitives *)
```

```
l1 = Table[Line[{{1, x}, {1 - x, 1}}], {x, 0, 1, 0.05}];
bild = Graphics[l1, AspectRatio → 1]
```



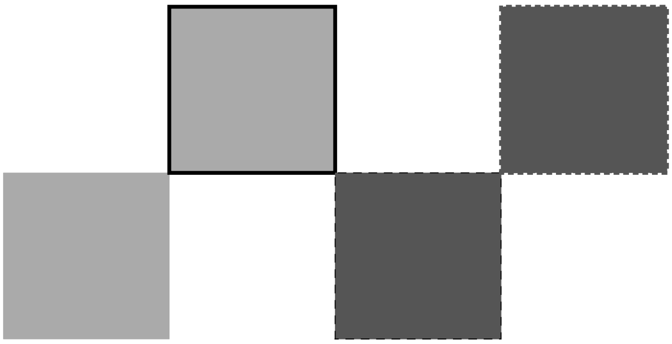
```
(* Farben und EdgeForm -man beachte ineinandergeschachtelte Listen  *)
```

```
g = Graphics[{Yellow, Rectangle[{0, 0}],
   {EdgeForm[Thick], Rectangle[{1, 1}]},
   Red, {EdgeForm[Dashed], Rectangle[{2, 0}]},
   {EdgeForm[{Thick, Dotted, Green}], Rectangle[{3, 1}]}}]
```



```
(* nun in Grau *)
```

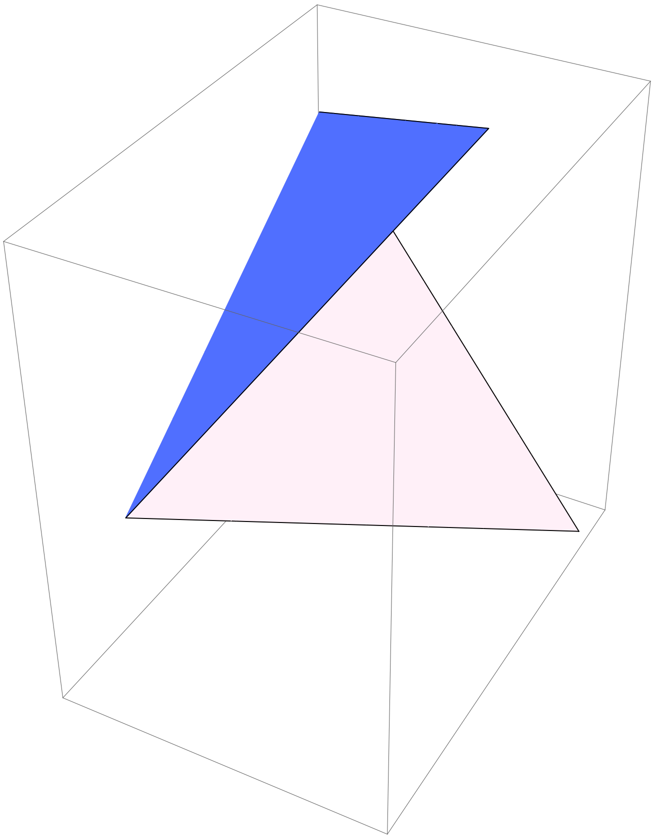
```
g /. RGBColor[a__] :→ GrayLevel[Mean[{a}]]
```



```
(*  3D Bilder Zusammenbauen *)
```

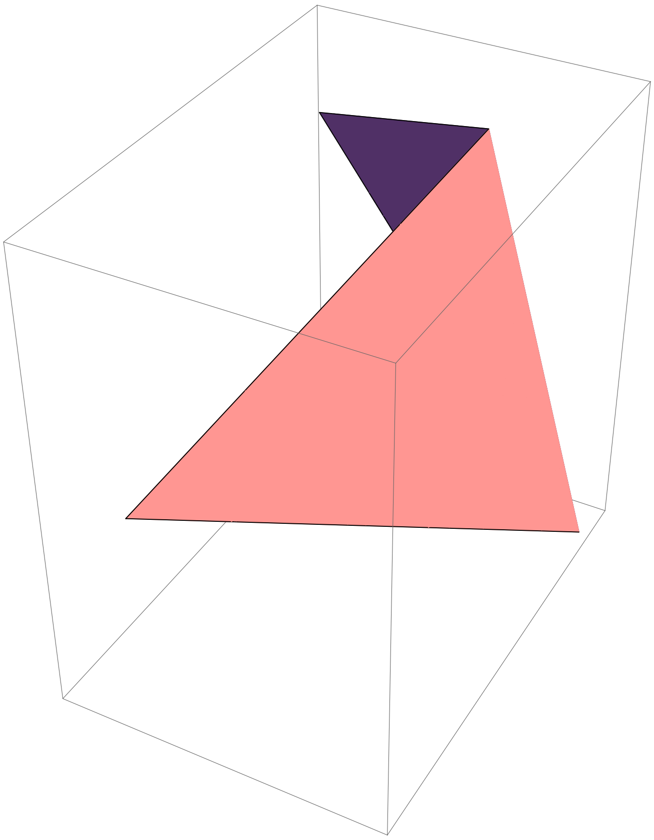
```
randpoints[n_] := Table[Random[], {n}]
```

```
p = Table[randpoints[3], {4}];
Graphics3D[Polygon[p]]
```

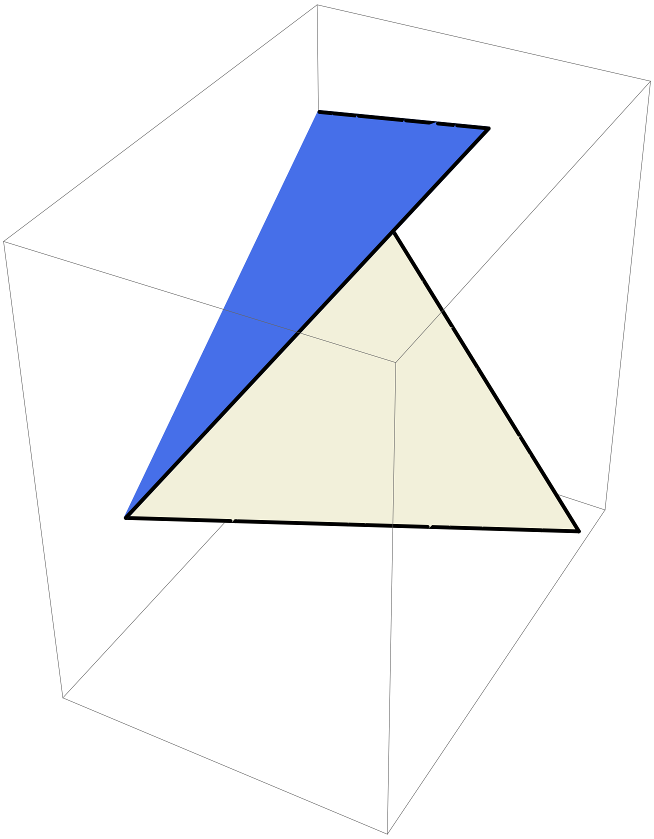


```
q = RotateLeft[p]
Graphics3D[Polygon[q]]
```

```
{{0.829756, 0.889715, 0.111417}, {0.236236, 0.973951, 0.79687},
 {0.666503, 0.751887, 0.97614}, {0.377839, 0.11144, 0.537799}}
```

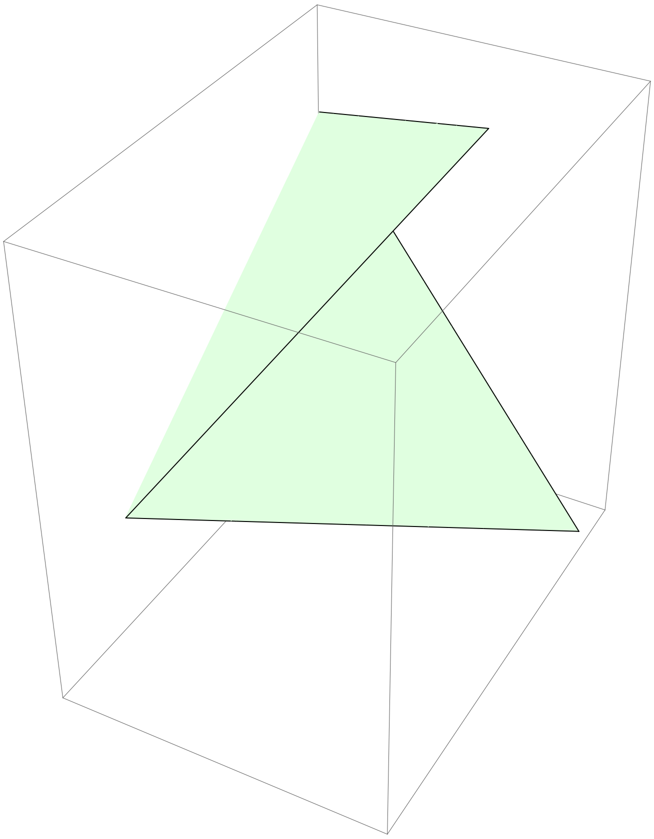


```
Graphics3D[{EdgeForm[Thick], FaceForm[LightGreen], Polygon[p]}]
```



```
(* die Einstellung Gree wirkt nicht,
weil das allgemeine Beleuchtungskonzep von Mma uebergeordnet ist *)
```

```
Graphics3D[{FaceForm[LightGreen], Polygon[p]}, Lighting → {White}]
```

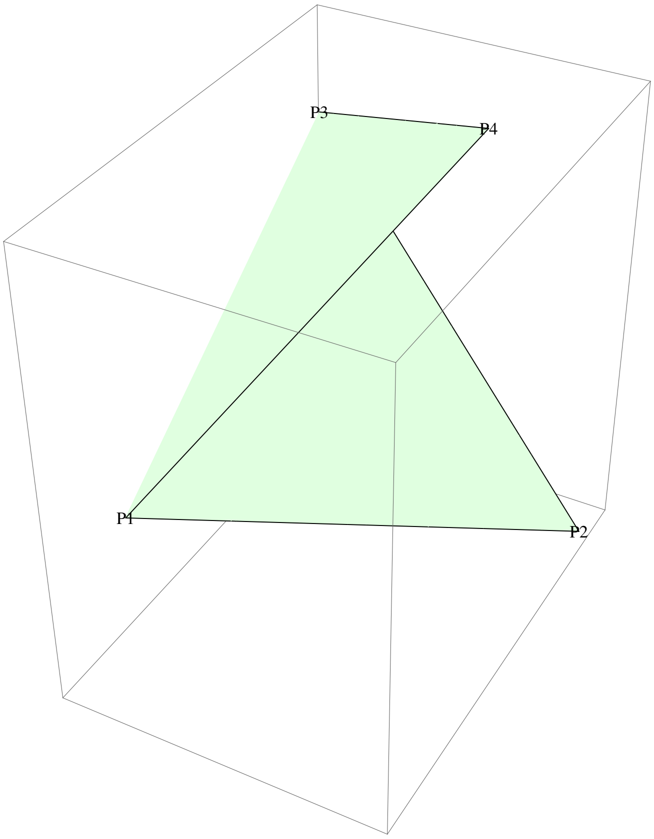


```
(* Beschriftung der Grafik *)
text = {P1, P2, P3, P4}
ZeigeBild :=
  Graphics3D[{schrift, {FaceForm[LightGreen], Polygon[p]}}, Lighting → {White}];
```

```
{P1, P2, P3, P4}
```

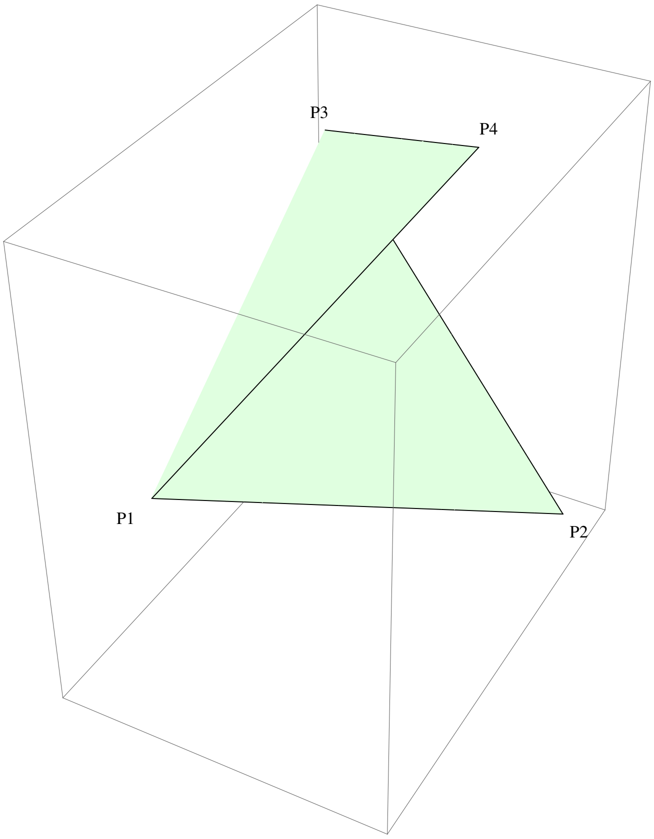
```
schrift = Table[Text[text[[i]], p[[i]] ] , {i, 1, 4}];
```

```
ZeigeBild
```



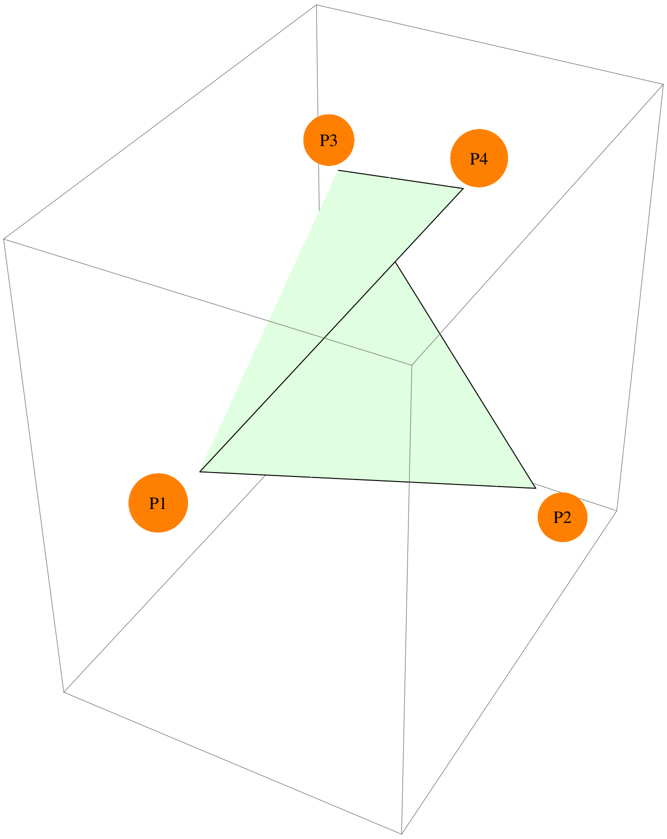
```
(* die Pi kleben noch auf den Ecken *)

schw = Mean[p];
schrift = Table[Text[text[[i]], 1.1 p[[i]] - 0.1 schw ] , {i, 1, 4}];
```

```
ZeigeBild
```



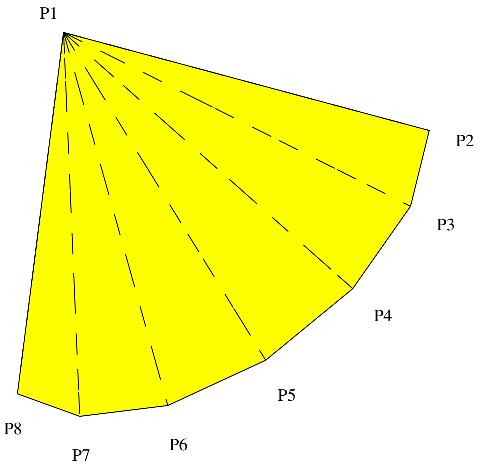
```
schrift = Table[With[{pos = 1.2 p[[i]] - 0.2 schw},
     {Text[text[[i]], pos ] , {Orange, Sphere[pos, 0.06]}}], {i, 1, 4}];
```

```
ZeigeBild
```

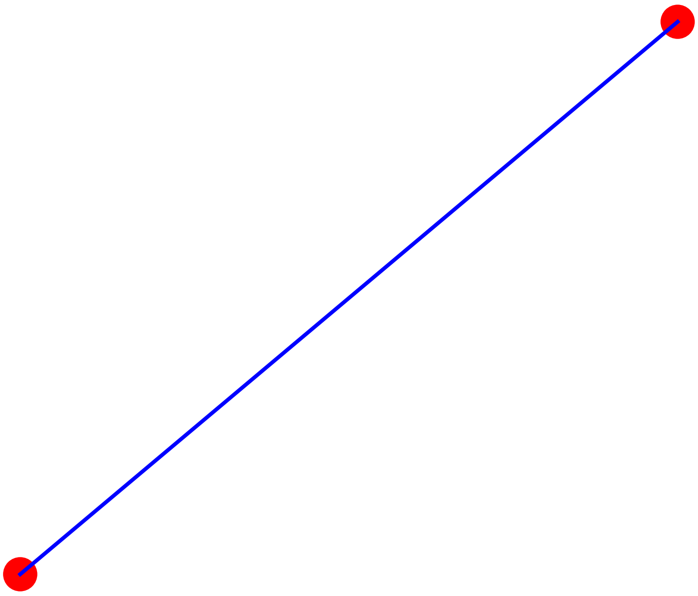


```
(* Bei n Punkten spielt P_1 die zentrale Rolle
  Polygon[P1,...,Pn] wird aus Dreiecken P_1P_iP_i+1 gebildet *)

p = Prepend[Table[{Sin[i], Cos[i], 0}, {i, 1.5, 4, 0.4}], {0, 0, 1}]

{{0, 0, 1}, {0.997495, 0.0707372, 0}, {0.9463, -0.32329, 0},
 {0.745705, -0.666276, 0}, {0.42738, -0.904072, 0}, {0.0415807, -0.999135, 0},
 {-0.350783, -0.936457, 0}, {-0.687766, -0.725932, 0}}

n = Length[p];
txt = Table["P" <> ToString[i], {i, 1, n}]

{P1, P2, P3, P4, P5, P6, P7, P8}

lines = Table[Line[{p[[1]], p[[i]]}], {i, 2, n}];

schrift =
  Table[Text[txt[[i]], If[i > 1, 1.1 p[[i]] - 0.1 p[[1]], {0, 0, 1.1}]], {i, 1, n}];
```

```
Graphics3D[{schrift, {Dashing[0.03], lines},
   {FaceForm[Yellow], Polygon[p]}},
 Lighting → {White}, PlotRange → All, Boxed → False]
```

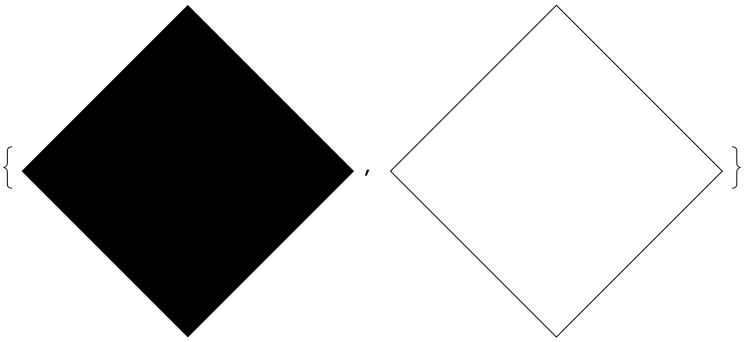


```
(* Zusammensetzung von verschiedenen
 GraphicsPrimitives in einem  GraphicsComplex *)

gc = GraphicsComplex[{{1.2, 3.5}, {5.6, 7.2}},
   {{Red, Point[1], Point[2]}, Line[{1, 2}]}]

GraphicsComplex[{{1.2, 3.5}, {5.6, 7.2}},
 {{RGBColor[1, 0, 0], Point[1], Point[2]}, Line[{1, 2}]}]

Graphics[{Blue, PointSize[0.05], Thick, gc}]
```



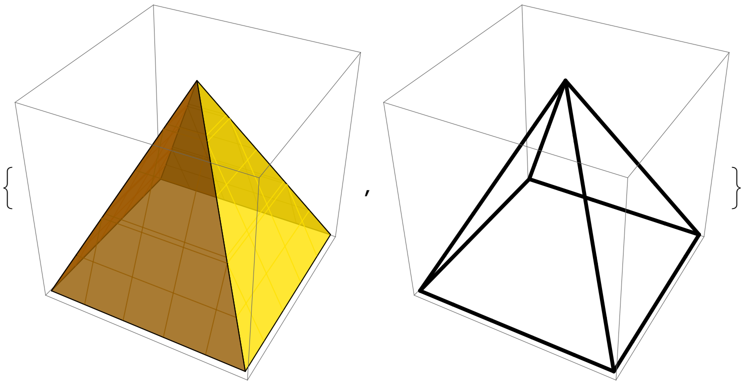
```
v = {{1, 0}, {0, 1}, {-1, 0}, {0, -1}};
```

```
{Graphics[GraphicsComplex[v, Polygon[{1, 2, 3, 4}]]],
 Graphics[GraphicsComplex[v, Line[{1, 2, 3, 4, 1}]]]}
```



```
v = {{0, 0, 0}, {2, 0, 0}, {2, 2, 0}, {0, 2, 0}, {1, 1, 2}};
i = {{1, 2, 5}, {2, 3, 5}, {3, 4, 5}, {4, 1, 5}};
```

```
{Graphics3D[{Opacity[.8], Yellow, GraphicsComplex[v, Polygon[i]]}],
 Graphics3D[{Thick, GraphicsComplex[v, Line[i]]}]}
```



Use built - in PolyhedronData :

```
v = PolyhedronData["Dodecahedron", "VertexCoordinates"];
Short[i = PolyhedronData["Dodecahedron", "FaceIndices"]]
```

```
{{15, 10, 9, 14, 1}, {2, 6, 12, 11, 5}, <<8>>, {3, 7, 16, 1, 14}, {16, 8, 4, 15, 1}}
```

```
{Graphics3D[{Yellow, GraphicsComplex[v, Polygon[i]]}],
 Graphics3D[{Thick, GraphicsComplex[v, Line[i]]}]}
```