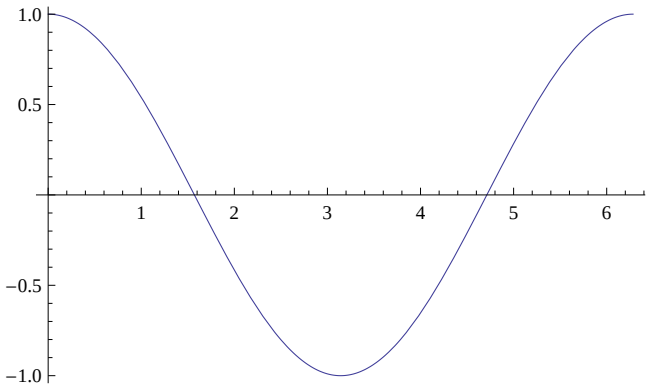
KMO_1 04

1. Podstawy sporzadzania wykresow

In[1]:=
```
Plot[Cos[x], {x, 0, 2 π}]
```

Out[1]=



In[2]:=
```
Plot[{Cos[x], Cos[2 x], Cos[3 x]}, {x, 0, 2 π}]
```

Automatyzacja :

In[4]:=
```
Plot[Table[BesselJ[n, x], {n, 5}], {x, -10, 10}]
```

Przedstawianie rozwiazania rownan rozniczkowych

In[16]:=
```
s = NDSolve[{y'[x] == y[x], y[0] == 1}, y, {x, 0, 4}]
```

Out[16]=
```
{{y → InterpolatingFunction[{{0., 4.}}, <>]}}
```

Evaluate[f] - rozwija funkcje f

In[13]:=
```
Plot[Evaluate[y[x] /. s], {x, 0, 4}, PlotRange → All]
```

In[18]:=
```
Evaluate[y[x] /. s][[1]]
```

Out[18]=
```
InterpolatingFunction[{{0., 4.}}, <>][x]
```

In[14]:=
```
s = NDSolve[{y'[x] == y[x] Cos[x + y[x]], y[0] == 1}, y, {x, 0, 30}]
```

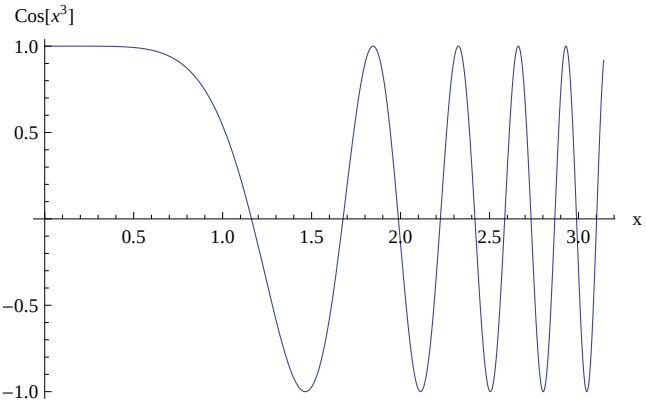
Out[14]=
```
{{y → InterpolatingFunction[{{0., 30.}}, <>]}}
```

In[15]:=
```
Plot[Evaluate[y[x] /. s], {x, 0, 30}, PlotRange → All]
```

2. Opcje instrukcji graficznych

In[22]:=
```
Plot[Cos[x^3], {x, 0, π}, AxesLabel → {"x", "Cos[x^3]"}]
```

Out[22]=



In[23]:=
```
Plot[Cos[x^3], {x, 0, π}, Frame → True, GridLines → Automatic]
```

In[27]:=
```
Plot[Cos[x^3], {x, 0, π}, AspectRatio → Automatic, ImageSize → {300, 150}]
```

In[30]:=
```
Plot[Sin[x], {x, 0, 2 π}, BaseStyle → {FontSize -> 18, FontFamily → "Times", Italic}]
```

In[32]:=
```
Plot[Sin[x], {x, 0, 2 π}, Ticks →
  {{{π / 2, "π/2"}, {π, "π"}, {3 π / 2, "3π/2"}, {2 π, "2π"}}, Automatic (*Os Y bez zmian *)}]
```

In[38]:=
```
Plot[Sin[1 / x], {x, 0.022, 0.3}, PlotRange → {0, 1}]
```

PlotPoints

okresla minimalna liczbe punktow, w ktorych nastepuje obliczenie wartosci funkcji; wartosc domyslna to 25

MaxRecursion

okresla maksymalna liczbe punktow podpodzialow poszczegolnych przedzialow (wyznaczonych przez PlotPoints)

In[44]:=
```
Plot[Sin[1 / x], {x, 0.022, 0.3}, MaxRecursion → 1]
```

In[45]:=
```
Plot[Sin[1 / x], {x, 0.022, 0.3}, MaxRecursion → 1, PlotPoints → 100]
```

In[51]:=
```
Plot[Sin[x], {x, -π/2, 5/2 π}, PlotStyle → {Hue[0.656], Thickness[0.01],
  Dashing[{0.03, 0.05}(*dlugosc kreski, dlugosc przerwy*)]}]
```

Hue[b, n, j] - barwa, nasycenie, jasnosc

Hue[b] == Hue[b, 1, 1]

In[52]:=
```
Plot[{Sin[x], Cos[x]}, {x, -π/2, 5/2 π},
 PlotStyle → {{Thickness[0.001], CMYKColor[0, 0.1, 0.2, 0.3]},
   {Thickness[0.02], RGBColor[0.7, 0, 0]}}]
```

In[54]:=
```
Plot[{Sin[x], Cos[x]}, {x, -π/2, 5/2 π},
 PlotStyle → {{Thickness[0.001], GrayLevel[0.25]}, {Thickness[0.02], GrayLevel[0.75]}}]
```

3. Maipulacja opcjami

In[117]:=
```
Options[Plot, PlotRange]
```

Out[117]=
```
{PlotRange → {Full, Automatic}}
```

In[120]:=
```
SetOptions[Plot, PlotRange → All]
```

Out[120]=
```
{AlignmentPoint → Center, AspectRatio → 1/GoldenRatio, Axes → True,
 AxesLabel → None, AxesOrigin → Automatic, AxesStyle → {}, Background → None,
 BaselinePosition → Automatic, BaseStyle → {}, ClippingStyle → None,
 ColorFunction → Automatic, ColorFunctionScaling → True, ColorOutput → Automatic,
 ContentSelectable → Automatic, CoordinatesToolOptions → Automatic,
 DisplayFunction :→ $DisplayFunction, Epilog → {}, Evaluated → Automatic,
 EvaluationMonitor → None, Exclusions → Automatic, ExclusionsStyle → None, Filling → None,
 FillingStyle → Automatic, FormatType :→ TraditionalForm, Frame → False, FrameLabel → None,
 FrameStyle → {}, FrameTicks → Automatic, FrameTicksStyle → {}, GridLines → None,
 GridLinesStyle → {}, ImageMargins → 0., ImagePadding → All, ImageSize → Automatic,
 ImageSizeRaw → Automatic, LabelStyle → {}, MaxRecursion → Automatic, Mesh → None,
 MeshFunctions → {#1 &}, MeshShading → None, MeshStyle → Automatic, Method → Automatic,
 PerformanceGoal :→ $PerformanceGoal, PlotLabel → None, PlotPoints → Automatic,
 PlotRange → All, PlotRangeClipping → True, PlotRangePadding → Automatic,
 PlotRegion → Automatic, PlotStyle → Automatic, PreserveImageOptions → Automatic,
 Prolog → {}, RegionFunction → (True &), RotateLabel → True,
 Ticks → Automatic, TicksStyle → {}, WorkingPrecision → MachinePrecision}
```

In[121]:=
```
Options[Plot, PlotRange]
```

Out[121]=
```
{PlotRange → All}
```

4. Powtarzanie wykresow

In[88]:=
```
Plot[ChebyshevU[5, x], {x, -1, 1}]
```

In[89]:=
```
Show[%]
```

In[90]:=
```
Show[%, PlotRange → {-1, 1}]
```

In[92]:=
```
Show[%, Axes → None, PlotLabel → "Wielomian Czebeszewa"]
```

In[93]:=
```
r1 = Plot[Gamma[1, x], {x, 0.1, 5}]
```

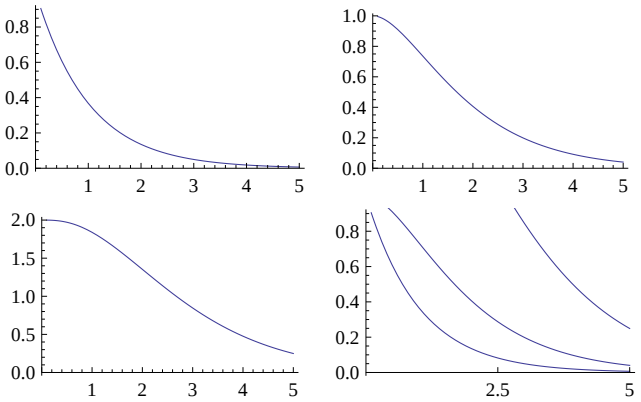
In[94]:=
```
r2 = Plot[Gamma[2, x], {x, 0.1, 5}]
```

In[95]:=
```
r3 = Plot[Gamma[3, x], {x, 0.1, 5}]
```

In[96]:=
```
r4 = Show[r1, r2, r3, Ticks → {{2.5, 5}, Automatic}]
```

In[103]:=
```
r5 = Show[GraphicsArray[{{r1, r2}, {r3, r4}}]]
```

Out[103]=



5. wykresy warstwicowe i cieniowane

In[125]:=
```
ContourPlot[Sin[x] Cos[y], {x, -2, 7}, {y, -7, 2}]
```

In[129]:=
```
ContourPlot[Sin[x] Cos[y], {x, -2, 7}, {y, -7, 2}, ContourShading → None]
```

In[128]:=
```
ContourPlot[Sin[x] Cos[y], {x, -2, 7}, {y, -7, 2},
 ContourShading → Automatic, ColorFunction → "Rainbow"]
```

In[133]:=
```
ContourPlot[Sin[x] Cos[y], {x, -2, 7}, {y, -7, 2},
 Contours → 3, ContourShading → {Red, Orange, Yellow, White}]
```

In[134]:=
```
DensityPlot[Sin[x] Cos[y], {x, -2, 7}, {y, -7, 2}]
```

In[135]:=
```
DensityPlot[Sin[x] Cos[y], {x, -2, 7}, {y, -7, 2}, Mesh → True]
```

In[136]:=
```
st[f_] := If[Mod[f^2, 2] > 0.5, Hue[0.15], Hue[Random[]]]
```

In[160]:=
```
DensityPlot[{Sin[x^2 + y^2], Sin[x^2 + y^2]}, {x, -2, 2},
 {y, -2, 2}, PlotPoints → 30, ColorFunction → (Hue[Sin[(#1)^2]] &)]
```

In[149]:=
```
ff[x_] := (x + 1)^2
```

In[151]:=
```
ff[4]
```

Out[151]= 25

In[152]:=
```
(#1 + 1)^2 &
```

Out[152]= $(\#1 + 1)^2 \&$

In[153]:=
```
%[4]
```

Out[153]= 25

6. Wykresy warstwicowe funkcji trzech zmiennych

Hiperboloidy jednopowlokowe :

In[28]:=
```
ContourPlot3D[x^2 - y^2 + z^2, {x, -3, 3},
 {y, -3, 3}, {z, -7, 7}, Contours → {0, 8, 16}(*poziomice *),
 ContourStyle → {{RGBColor[1, 0, 0]}, {RGBColor[1, 1, 0]}, {RGBColor[1, 0, 1]}}
 (*,Lighting→None*), PlotPoints → {7, 2, 7}(*odpowiedno dla x, y, i z*)]
```

In[30]:=
```
tabf = Table[x^2 - y^2 + z^2, {x, -3, 3}, {y, -3, 3}, {z, -7, 7}];
```

In[31]:=
```
ListContourPlot3D[tabf, Contours → {0, 8, 16}]
```

7. Wykreslanie listy danych

In[161]:=
```
t = Table[i^3, {i, 10}]
```

Out[161]=
```
{1, 8, 27, 64, 125, 216, 343, 512, 729, 1000}
```

In[162]:=
```
ListPlot[t, PlotStyle → {PointSize[0.05]}]
```

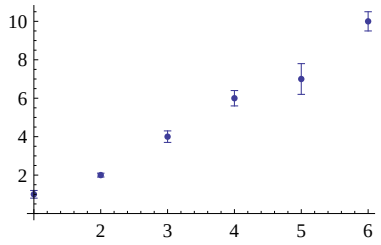
In[164]:=
```
ListPlot[t, Joined -> True, PlotStyle -> {Thickness[0.01]}]
```

8. Zazanczanie bledow danych

In[1]:=
```
Needs["ErrorBarPlots`"]
```

In[8]:=
```
ErrorListPlot[{{{1, 1}, ErrorBar[0.2]}, {{2, 2}, ErrorBar[0.1]}, {{3, 4}, ErrorBar[0.3]},
  {{4, 6}, ErrorBar[0.4]}, {{5, 7}, ErrorBar[0.8]}, {{6, 10}, ErrorBar[0.5]}}]
```

Out[8]=

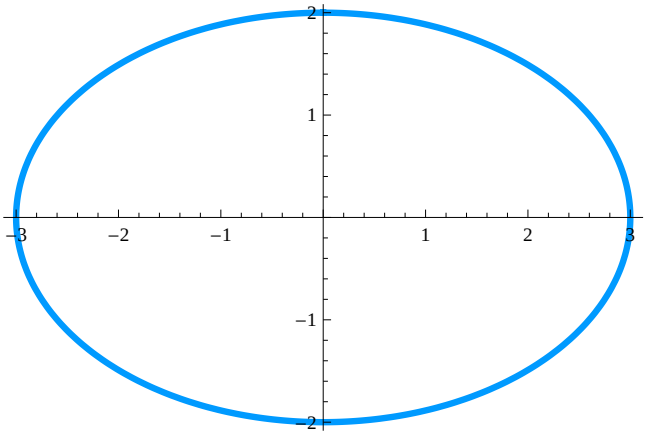


9. wykresy funkcji parametrycznych

In[165]:=
```
ParametricPlot[{(Cos[t])^3, (Sin[t])^3}, {t, 0, 2 π}]
```

In[167]:=
```
ParametricPlot[{3 Sin[t], 2 Cos[t]}, {t, 0, 2 π}, PlotStyle → {Hue[0.566], Thickness[0.01]}]
```

Out[167]=



In[169]:=
```
ParametricPlot3D[{3 Sin[t], 2 Cos[t], t / 4},
  {t, 0, 7 π}, PlotStyle → {Hue[0.566], Thickness[0.01]}]
```

In[173]:=
```
ParametricPlot3D[{u 3 Sin[t], u 2 Cos[t], t}, {t, 0, 7 π},
  {u, -1, 1}, PlotStyle → {Hue[0.566], Thickness[0.01]}, PlotPoints → 50]
```

In[178]:=
```
ParametricPlot3D[{Cos[t] Cos[u], Sin[t] Cos[u], Sin[u]}, {t, 0, 2 π}, {u, -π / 2, π / 2}]
```

In[181]:=
```
ParametricPlot3D[{Cos[t] (3 + Cos[u]), Sin[t] (3 + Cos[u]), Sin[u]}, {t, 0, 2 π}, {u, 0, 2 π}]
```

10. Animacje

In[182]:=
```
<< Graphics‘Animation‘
```

```
f[x_, y_, t_] := Sin[Sqrt[x^2 + y^2] - t] / Sqrt[x^2 + y^2]
```

```
Animate[Plot3D[f[x, y, t], {x, -20, 20}, {y, -20, 20}, PlotRange -> {All, All, {-2, 2}},
  PlotPoints -> 50, Mesh -> False, Boxed -> False, Axes -> False], {t, 0, 6, 0.3}]
```

In[185]:=
```
g[x_, y_, t_] := Cos[10 π Sqrt[x^2 + y^2] + 2 ArcTan[y, x] - t]
```

```
Animate[Plot3D[g[x, y, t], {x, -0.7, 0.7}, {y, -0.7, 0.7}, PlotRange -> {All, All, {-9, 9}},
  PlotPoints -> 100, Mesh -> False, Boxed -> False, Axes -> False], {t, 0, 6, 0.3}]
```

```
Show[%, ViewPoint -> {2, 2, 6}]
```

```
- SurfaceGraphics -
```

```
Table[
 ParametricPlot3D[{{Cos[θ + i] Sin[α] + 5 Cos[i], Sin[θ + i] Sin[α] - 5 Sin[i], Cos[α] - 2}},
  {α, 0, 2 π}, {θ, 0, 2 π}], {i, 0, 2 π, 0.3}]
```

11. Histogramy

In[188]:=
```
dane = {0.0, 2, 4, 4.3, 4.4, 4.45, 5, 7, 8.01, 8.51, 8.9, 9.91, 9.97, 9.99, 0.995};
```

In[192]:=
```
Histogram[dane, 4]
```

In[209]:=
```
data1 = RandomReal[NormalDistribution[0, 0.5], 1000];
data2 = RandomReal[NormalDistribution[3, 1 / 2], 1000];
```

In[211]:=
```
Histogram[{data1, data2}, 64]
```

12. Pola wektorowe

In[10]:=
```
GradientFieldPlot[f_, {x_, xmin_, xmax_}, {y_, ymin_, ymax_}, opts : OptionsPattern[]] :=
 VectorPlot[Evaluate[D[f, {{x, y}}]], {x, xmin, xmax}, {y, ymin, ymax}, opts]
```

In[11]:=
```
GradientFieldPlot[Sqrt[x^2 + y^2], {x, -5, 5}, {y, -5, 5}]
```

In[12]:=
```
VectorPlot[{Sin[x], Cos[y]}, {x, 0, 2 Pi}, {y, 0, 2 Pi}, VectorColorFunction → Hue]
```

Opracowano na podstawie ksiazki Grzegorz Drwal, (+inni) - "Mahematica 5"

mgr Andrzej Pisarski