Ruch

Położenie x(t), Prędkość V(t), Przyspieszenie a(t)

E8. A particle moves along a straight line with equation of motion $s =$
seconds. Find the velocity and the speed when $t = 5$ for:

a. $f(t) = 150 + 50\,t - 4.5\,t^2$,

b. $f(t) = t^{-1} - t$.

Podpunkt a)

```
x = 150 + 50 t - 4.5 t^2
```

$150 + 50\,t - 4.5\,t^2$

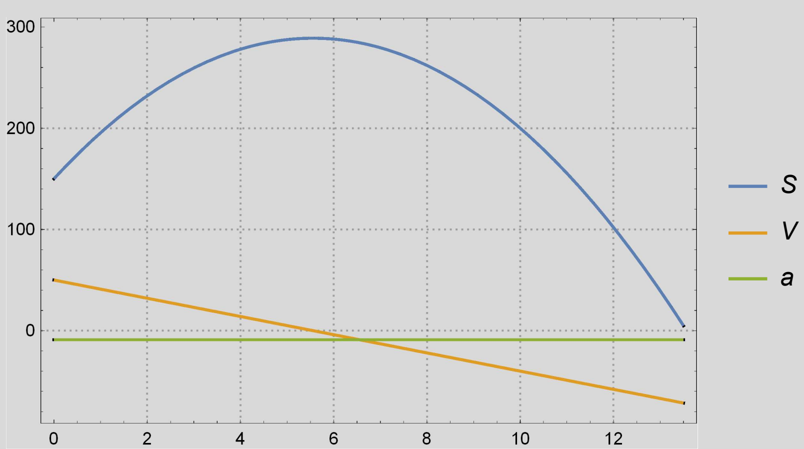
```
V = D[x, t]
```

```
50 - 9. t
```

```
a = D[V, t]
```

```
-9.
```

```
Plot[{x, V, a}, {t, 0, 13.5}, PlotTheme → "Detailed"]
```



Podpunkt b)

```
x = 1/t - t
```

$\frac{1}{t} - t$

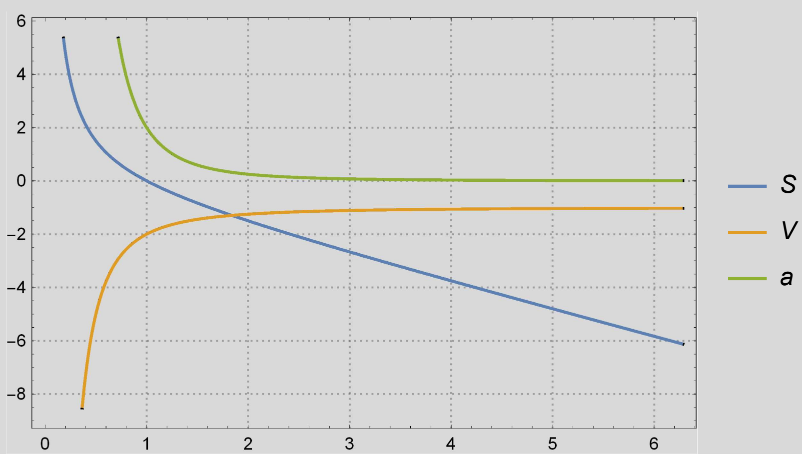
```
V = D[x, t]
```

$-1 - \frac{1}{t^2}$

```
a = D[V, t]
```

$\frac{2}{t^3}$

```
Plot[{x, V, a}, {t, 0, 2 π}, PlotTheme → "Detailed"]
```



Podpunkt c)

```
x = -Cos[t]
```

-Cos[t]

```
V = D[x, t]
```

Sin[t]

```
a = D[V, t]
```

Cos[t]

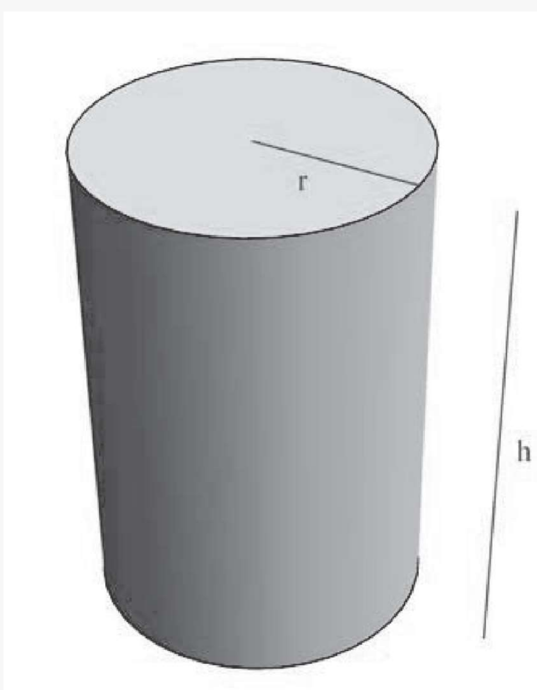
```
Plot[{x, V, a}, {t, 0, 2 π}, PlotTheme → "Detailed"]
```


```
Clear[x, V, a]
```

WAŻNE: Dlaczego pochodna w ogóle równa się zero?
Ciągłość a różniczkowalnie.

Optymalne projektowanie

Cylindryczna puszka ma pomieścić 1 litr oleju. **Znajdź optymalne wymiary puszki by koszt uzytego materiału był najmniejszy**.



V=1 litr objętości = 1000 cm^3

P= Pole powierzchni całej puszki = 2 * (Pole Podstawy) + Powierzchnia Boczna
P= 2 (πr^2) + 2 π r h

V=πr^2 h
a zatem h= $\frac{V}{\pi r^2}$
i możemy zapisać

```
P[r] = 2 (π r^2) + 2 π r V/(π r^2)
```

$2\pi r^2 + \frac{2V}{r}$

```
P' = D[P[r], r]
```

$4\pi r - \frac{2V}{r^2}$

$4\pi r - \frac{2V}{r^2} = 0$
$r^3 = \frac{V}{2\pi}$
$r = \left(\frac{V}{2\pi}\right)^{\frac{1}{3}}$

```
r = N[(1000/(2 π))^(1/3)]
```

5.41926

$$h = \frac{V}{\pi r^2}$$
$$r = \left(\frac{V}{2\pi}\right)^{\frac{1}{3}}$$

Jak podać wartość h w zaleznosci od r? $h = \frac{2V}{2\pi r^2} = \frac{V}{2\pi}\frac{2}{r^2}$ więc $\frac{V}{2\pi}$=h $\frac{r^2}{2}$

$$r = \left(\frac{V}{2\pi}\right)^{\frac{1}{3}} = \left(h\frac{r^2}{2}\right)^{\frac{1}{3}}$$
$$r^3 = \frac{h}{2} r^2$$
$$r = \frac{h}{2}$$

albo h =2r (czyl wysokość to tyle co średnica!)

Znaleźć najwiekszy wyraz ciągu a_n

$$a_n = \frac{2\,n}{n^2 + 100}$$

$$D\left[\frac{2\,x}{x^2 + 100}, x\right]$$

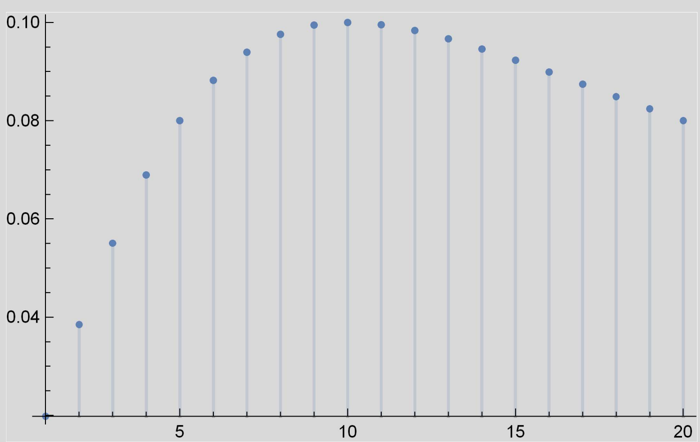
$$-\frac{4\,x^2}{\left(100 + x^2\right)^2} + \frac{2}{100 + x^2}$$

$$\frac{2\left(x^2 + 100\right) - 2\,x\,(2\,x)}{\left(x^2 + 100\right)^2} = \frac{2\left(100 - x^2\right)}{\left(x^2 + 100\right)^2} = \frac{2\,(10 - x)\,(10 + x)}{\left(x^2 + 100\right)^2}$$

czyli n = 10

$$\text{Solve}\left[D\left[\frac{2\,x}{x^2 + 100}, x\right] == 0, x\right]$$

$$\text{DiscretePlot}\left[\frac{2\,n}{n^2 + 100}, \{n, 1, 20\}\right]$$



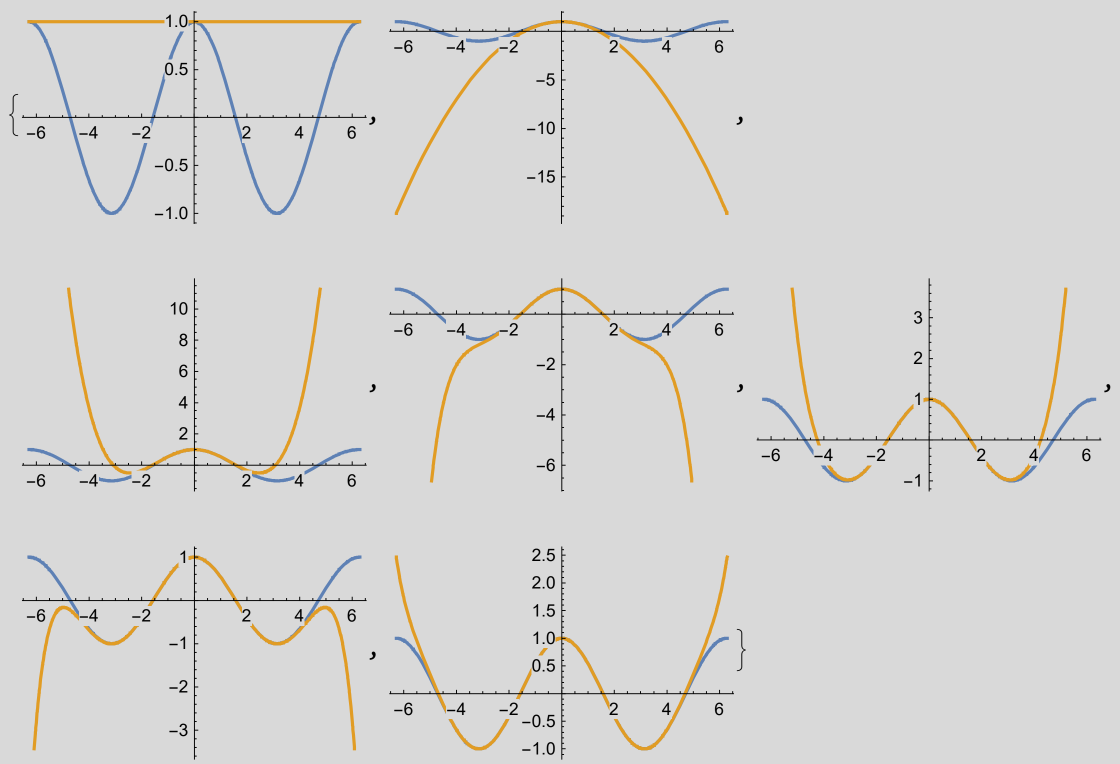
Szereg Taylora

Cos w $x_0 = 0$

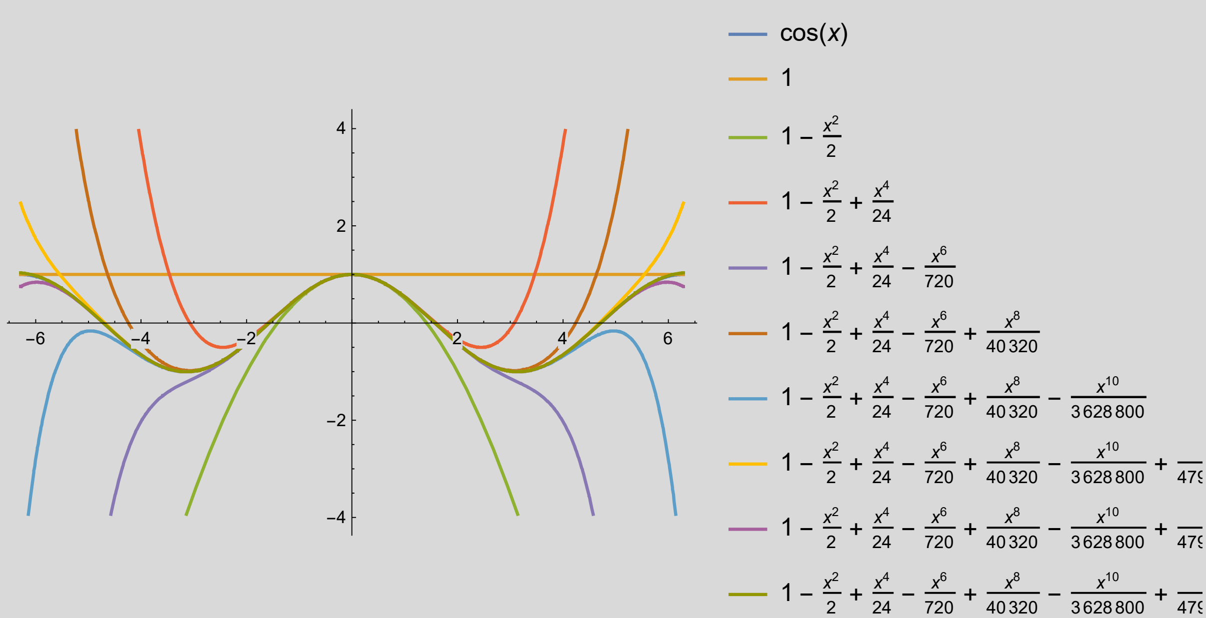
In[178]:=

```
polycos = Table[Normal[Series[Cos[x], {x, 0, k}]], {k, 1, 18, 2}];
Table[Plot[Evaluate[{Cos[x], polycos[[i]]}], {x, -2 π, 2 π}], {i, 1, 7}]
Plot[Evaluate[{Cos[x], polycos}], {x, -2 π, 2 π}, PlotLegends → "Expressions"]
```

Out[179]=



Out[180]=



Sinus w $x_0 = 0$

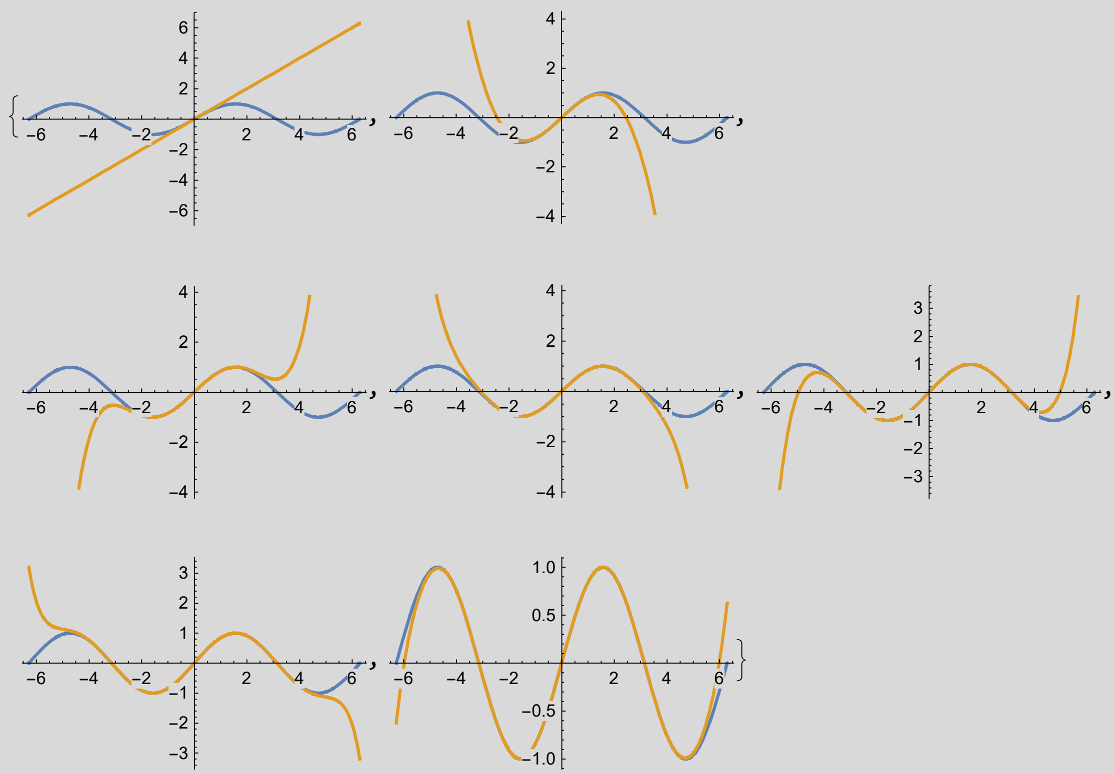
In[174]:=

```
polys = Table[Normal[Series[Sin[x], {x, 0, k}]], {k, 1, 14, 2}];
polys // TableForm
Table[Plot[Evaluate[{Sin[x], polys[[i]]}], {x, -2 π, 2 π}], {i, 1, 7}]
Plot[Evaluate[{Sin[x], polys}], {x, -2 π, 2 π}]
```

Out[175]//TableForm=

$$x$$
$$x - \frac{x^3}{6}$$
$$x - \frac{x^3}{6} + \frac{x^5}{120}$$
$$x - \frac{x^3}{6} + \frac{x^5}{120} - \frac{x^7}{5040}$$
$$x - \frac{x^3}{6} + \frac{x^5}{120} - \frac{x^7}{5040} + \frac{x^9}{362\,880}$$
$$x - \frac{x^3}{6} + \frac{x^5}{120} - \frac{x^7}{5040} + \frac{x^9}{362\,880} - \frac{x^{11}}{39\,916\,800}$$
$$x - \frac{x^3}{6} + \frac{x^5}{120} - \frac{x^7}{5040} + \frac{x^9}{362\,880} - \frac{x^{11}}{39\,916\,800} + \frac{x^{13}}{6\,227\,020\,800}$$

Out[176]=



Out[177]=

In[115]:=

```
poly = Table[Normal[Series[Sin[x], {x, π, k}]], {k, 1, 6, 2}];
poly // TableForm
Plot[Evaluate[{Sin[x], poly}], {x, 0, 3 π}]
```

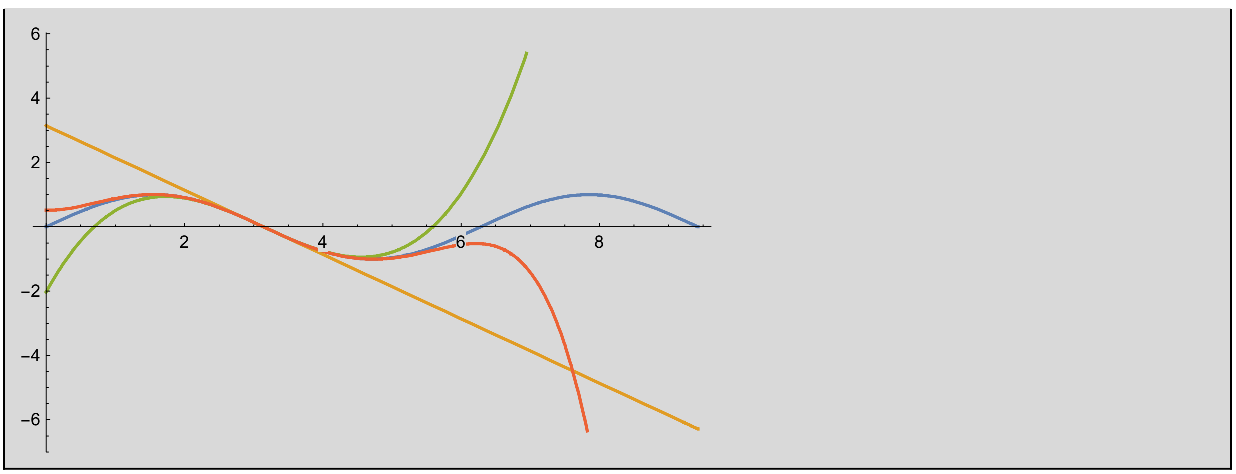
Out[116]//TableForm=

$\pi - x$

$\pi - x + \frac{1}{6}\ (-\pi + x)^3$

$\pi - x + \frac{1}{6}\ (-\pi + x)^3 - \frac{1}{120}\ (-\pi + x)^5$

Out[117]=



Analityczne rozwiązania w Mathematice Wolframa DSolve

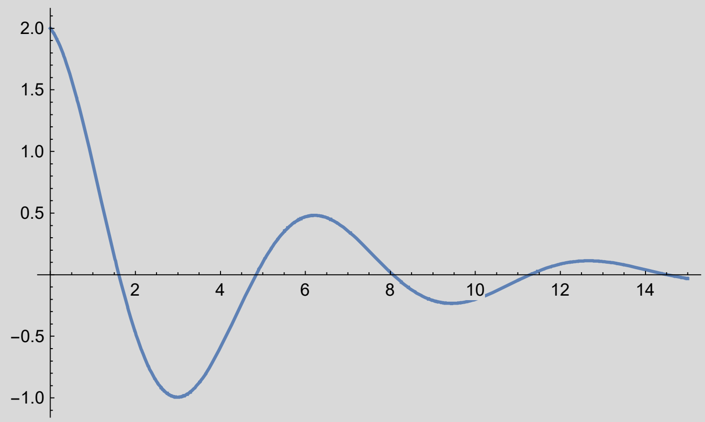
```
sol = DSolve[{y''[t] == -k y[t] - b y'[t] }, y[t], t]
```

$\left\{\left\{y[t] \to e^{\frac{1}{2}\left(-b-\sqrt{b^2-4 k}\right) t} C[1] + e^{\frac{1}{2}\left(-b+\sqrt{b^2-4 k}\right) t} C[2]\right\}\right\}$

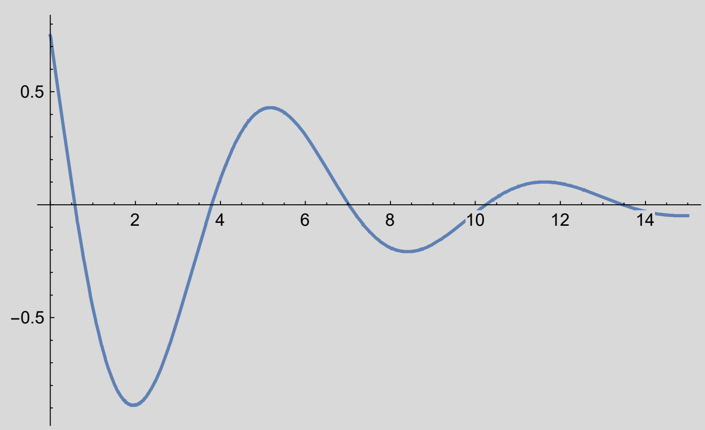
```
V = D[ⅇ^(-0.225` t) C[2] Cos[0.9743587634952539` t] +
    ⅇ^(-0.225` t) C[1] Sin[0.9743587634952539` t], t] // Simplify
```

```
ⅇ^(-0.225 t) ((0.974359 C[1] - 0.225 C[2]) Cos[0.974359 t] +
   (-0.225 C[1] - 0.974359 C[2]) Sin[0.974359 t])
```

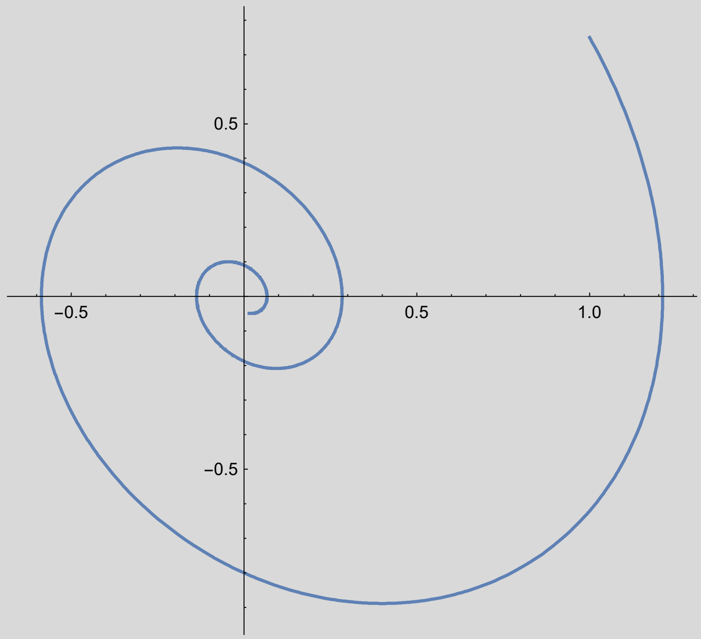
```
k = 1; b = 0.45; Plot[{{y[t]} /. sol /. {C[1] → 1, C[2] → 1}}, {t, 0, 15}]
```




```
k = 1; b = 0.45;
Plot[{{ⅇ^(-0.225` t) ((0.9743587634952539` C[1] - 0.225` C[2]) Cos[0.9743587634952539` t] +
        (-0.225` C[1] - 0.9743587634952539` C[2]) Sin[0.9743587634952539` t])} /.
    sol /. {C[1] → 1, C[2] → 1}}, {t, 0, 15}]
```



```
ParametricPlot[{y[t], y'[t]} /. sol /. {C[1] → 1, C[2] → 1}, {t, 0, 15}]
```



Ciało na sprężynie z oporem

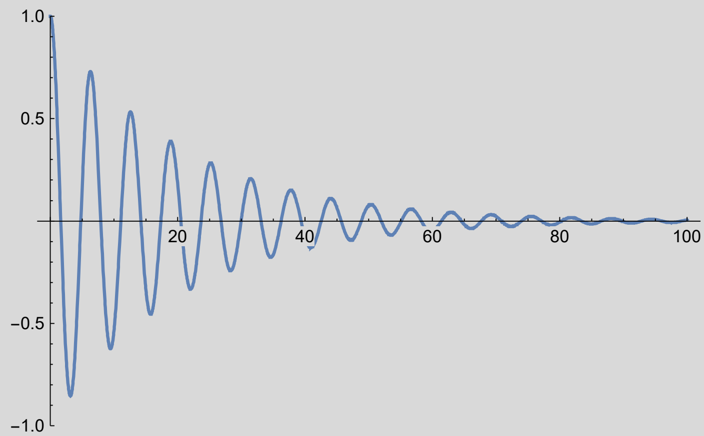
```
sol = DSolve[{x''[t] == -k/m x[t] - b/m x'[t], x[0] == 1, x'[0] == 0}, x[t], t] // Simplify
```

$$\left\{\left\{x[t] \to \frac{e^{-\frac{\left(b+\sqrt{b^2-4 k m}\right) t}{2 m}} \left(b \left(-1+e^{\frac{\sqrt{b^2-4 k m}\ t}{m}}\right)+\left(1+e^{\frac{\sqrt{b^2-4 k m}\ t}{m}}\right) \sqrt{b^2-4 k m}\right)}{2 \sqrt{b^2-4 k m}}\right\}\right\}$$

```
sol[[1]] // Flatten
```

$$\left\{x[t] \to \frac{e^{-\frac{\left(b+\sqrt{b^2-4 k m}\right) t}{2 m}} \left(b \left(-1+e^{\frac{\sqrt{b^2-4 k m}\ t}{m}}\right)+\left(1+e^{\frac{\sqrt{b^2-4 k m}\ t}{m}}\right) \sqrt{b^2-4 k m}\right)}{2 \sqrt{b^2-4 k m}}\right\}$$

```
Plot[x[t] /. sol /. {k → 1, m → 1, b → 1/10}, {t, 0, 100}, PlotRange → {-1, 1}]
```

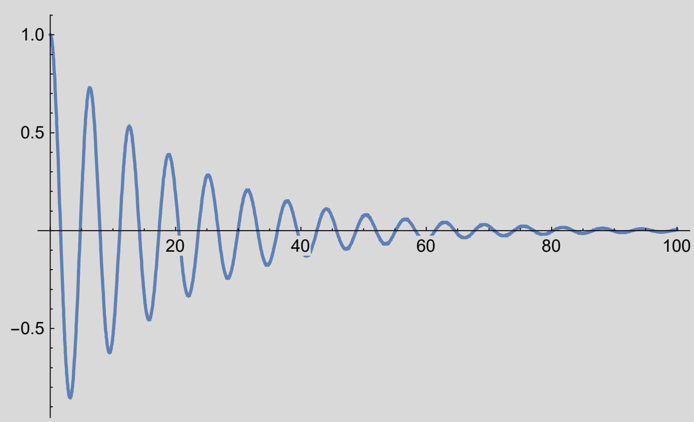



```
s = NDSolve[{x''[t] == -k/m x[t] - b/m x'[t], x[0] == 1, x'[0] == 0} /.
    {k → 1, m → 1, b → 1/10}, x, {t, 0, 100}]
```

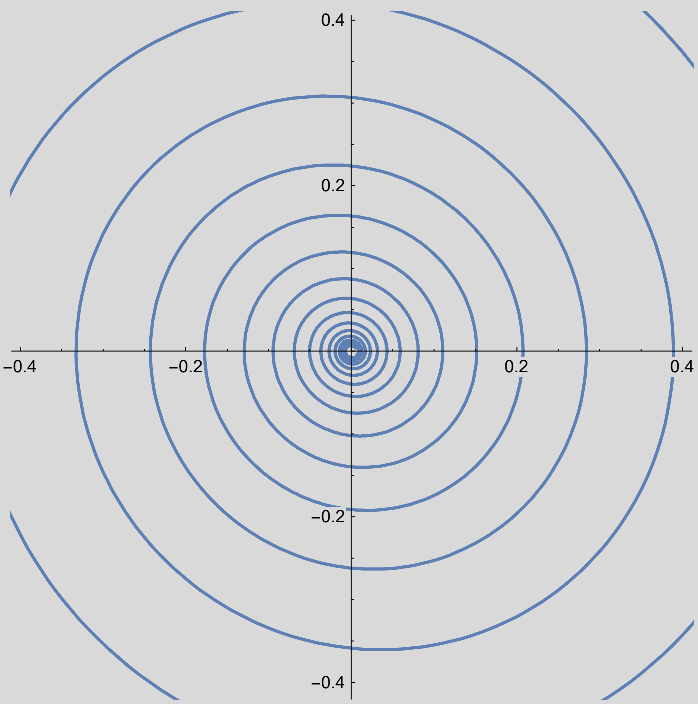
```
{{x → InterpolatingFunction[ Domain: {{0., 100.}} Output: scalar ]}}
```

Use the solution in a plot:

```
p1 = Plot[Evaluate[x[t] /. s], {t, 0, 100}, PlotRange → All]
```



```
ParametricPlot[Evaluate[{x[t], x'[t]} /. s], {t, 0, 100}]
```



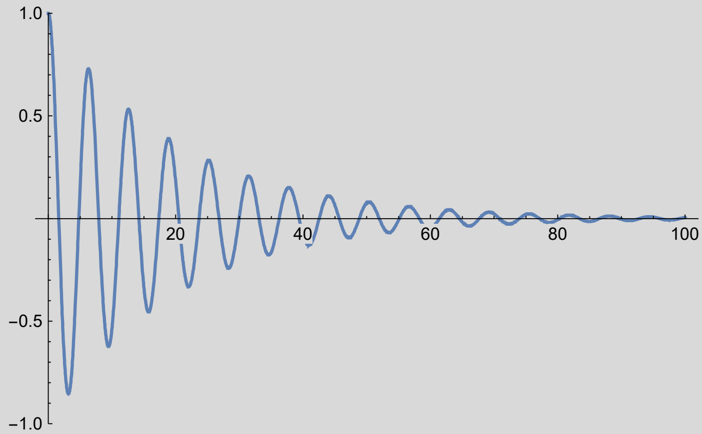

```
sol = DSolve[{x''[t] == -k/m x[t] - b/m x'[t], x[0] == 1, x'[0] == 0}, x[t], t] // Simplify
```

$$\left\{\left\{x[t] \to \frac{e^{-\frac{\left(b+\sqrt{b^2-4 k m}\right) t}{2 m}} \left(b \left(-1+e^{\frac{\sqrt{b^2-4 k m}\ t}{m}}\right)+\left(1+e^{\frac{\sqrt{b^2-4 k m}\ t}{m}}\right) \sqrt{b^2-4 k m}\right)}{2 \sqrt{b^2-4 k m}}\right\}\right\}$$

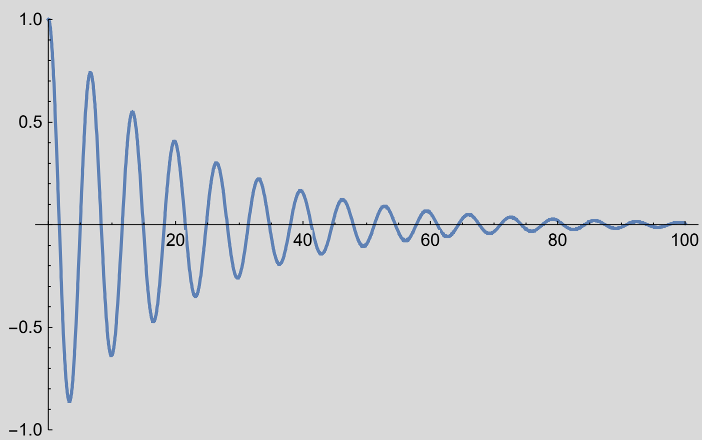
```
sol[[1]] // Flatten
```

$$\left\{x[t] \to \frac{e^{-\frac{\left(b+\sqrt{b^2-4 k m}\right) t}{2 m}} \left(b \left(-1+e^{\frac{\sqrt{b^2-4 k m}\ t}{m}}\right)+\left(1+e^{\frac{\sqrt{b^2-4 k m}\ t}{m}}\right) \sqrt{b^2-4 k m}\right)}{2 \sqrt{b^2-4 k m}}\right\}$$

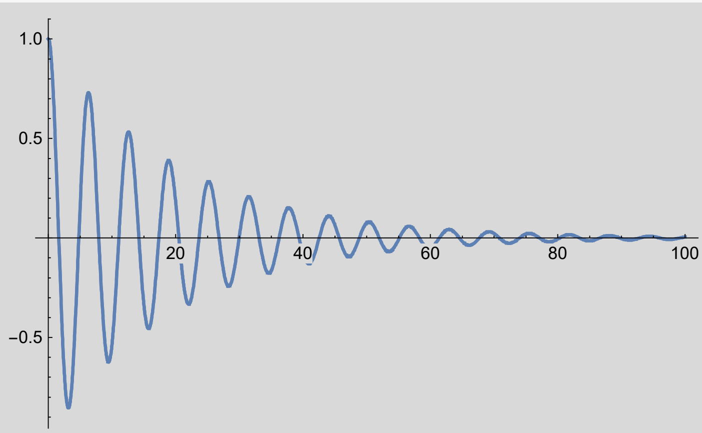
```
p2 = Plot[x[t] /. sol /. {k → 1, m → 1, b → 1/10}, {t, 0, 100}, PlotRange → {-1, 1}]
```




```
p3 = Plot[x[t] /. sol /. {k → 1, m → 1.1, b → 1/10}, {t, 0, 100}, PlotRange → {-1, 1}]
```

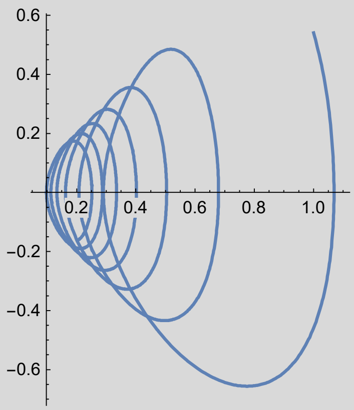



```
Show[p1, p2]
```



Use the function and its derivative in a plot:

```
ParametricPlot[Evaluate[{y[x], y'[x]} /. s], {x, 0, 20}]
```



DSolve

```
DSolve
```

```
DSolve[y'[x] + y[x] == a Sin[x], y[x], x]
```

$$\left\{\left\{y[x] \to e^{-x} C[1] + \frac{1}{2} a (-\text{Cos}[x] + \text{Sin}[x])\right\}\right\}$$

Include a boundary condition:

```
DSolve[{y'[x] + y[x] == a Sin[x], y[0] == 0}, y[x], x]
```

$$\left\{\left\{y[x] \to -\frac{1}{2} a e^{-x} \left(-1 + e^{x} \text{Cos}[x] - e^{x} \text{Sin}[x]\right)\right\}\right\}$$

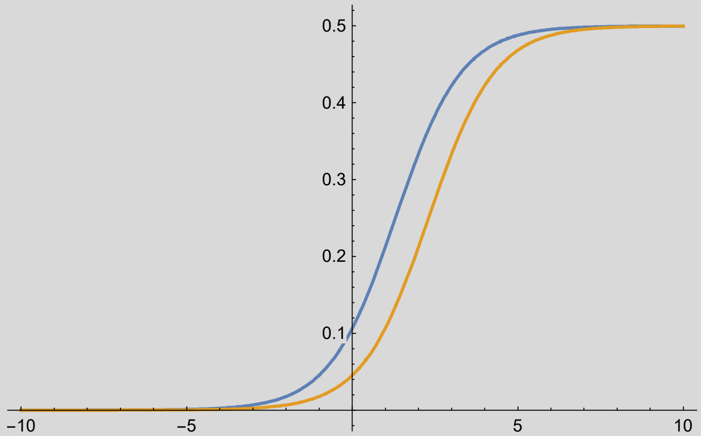
Plot the general solution of a differential equation:

```
sol = DSolve[{y'[x] == -y[x]^2 + y[x]}, y[x], x]
```

$$\left\{\left\{y[x] \to \frac{e^{x}}{e^{x} + e^{C[1]}}\right\}\right\}$$

Plot the solution curves for two different values of the arbitrary constant :

```
Plot[{y[x] /. sol /. C[1] → 2, y[x] /. sol /. C[1] → 3}, {x, -10, 10}]
```



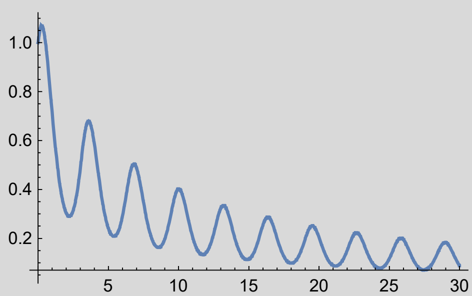
Solve a first-order ordinary differential equation:

```
s = NDSolve[{y'[x] == y[x] Cos[2 x + y[x]], y[0] == 1}, y, {x, 0, 30}]
```

```
{{y → InterpolatingFunction[ Domain: {{0., 30.}}
                             Output: scalar ]}}
```

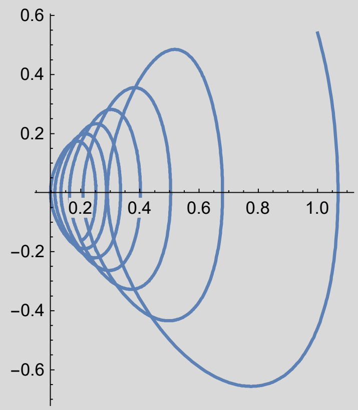
Use the solution in a plot:

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



Use the function and its derivative in a plot:

```
ParametricPlot[Evaluate[{y[x], y'[x]} /. s], {x, 0, 20}]
```

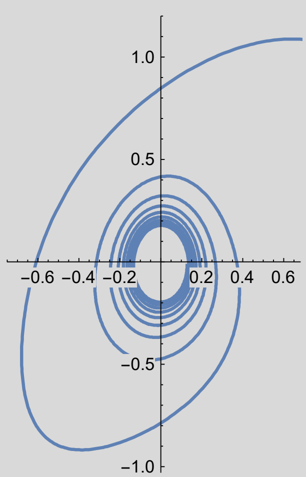


System of ordinary differential equations:

```
s = NDSolve[
  {x'[t] == -y[t] - x[t]^2, y'[t] == 2 x[t] - y[t]^3, x[0] == y[0] == 1}, {x, y}, {t, 40}]
```

```
{{x → InterpolatingFunction[ Domain: {{0., 40.}} Output: scalar ],
  y → InterpolatingFunction[ Domain: {{0., 40.}} Output: scalar ]}}
```

```
ParametricPlot[Evaluate[{x[t], y[t]} /. s], {t, 0, 40}]
```



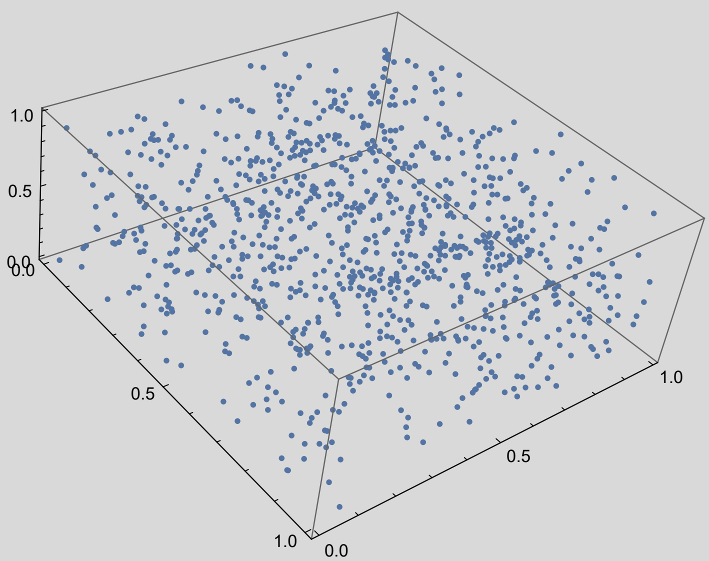
Trash

plot point

```
RandomReal[1, {6, 3}]
```

```
{{0.120123, 0.68727, 0.742396}, {0.139994, 0.42611, 0.455618},
 {0.203537, 0.657634, 0.957752}, {0.688981, 0.804038, 0.126622},
 {0.728338, 0.425736, 0.0173522}, {0.162898, 0.579071, 0.671466}}
```

```
ListPointPlot3D[RandomReal[1, {1000, 3}]]
```




```
RandomInteger[{1, 9}, 3]
```

```
{3, 2, 5}
```