Obyčejné diferenciální rovnice 1.řádu

I v tomto zdánlivě jednoduchém případě je nutné dívat pozor na výsledky - nemusejí být zcela dobře.

U obyčejných diferenciálních rovnic je jednoduchá geometricá interpretace, která velmi napomůže kontrole řešení. Je to tzv. směrové pole, tj. soubor vektorů, jejichž směrnice je v počátečním bodě (x,y) dána rovností y'=f(x,y), pokud je rovnice psána ve tvaru rozřešeném pro derivaci.

Lze použít buď příkaz VectorPlot nebo StreamPlot. Každý z těchto příkazů má své výhody a nevýhody.

```
odr1 = y'[x] - y[x] == Sin[x]
```

$$y'(x) - y(x) = \sin(x)$$

```
res1 = DSolve[rov1, y[x], x]
```

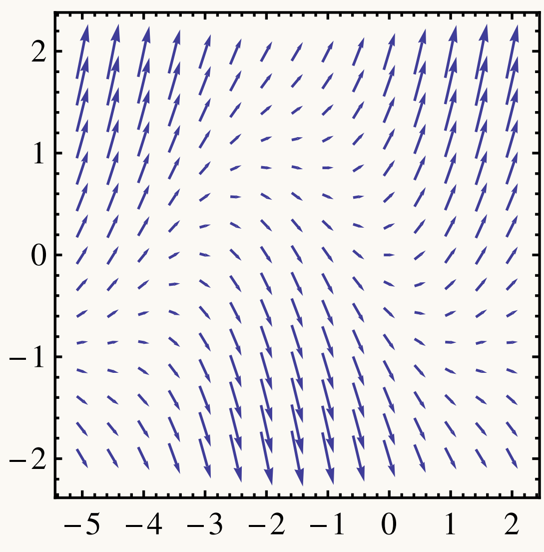
$$\left\{\left\{y(x) \to c_1 e^x + \frac{1}{2}(-\sin(x) - \cos(x))\right\}\right\}$$

Uvedeme nejdříve směrové pole pomocí VectorPlot a potom pomocí StreamPlot. Pak zvolíme jedno řešení a vepíšeme ho do směrového pole.

In[131]:=

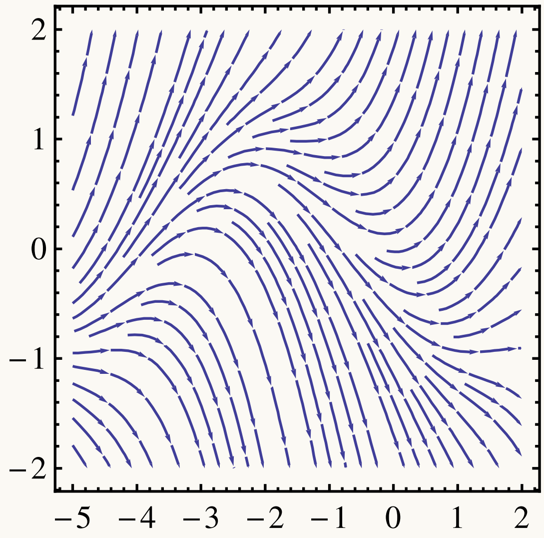
```
vp = VectorPlot[{1, y + Sin[x]}, {x, -5, 2}, {y, -2, 2},
  ImageSize → {200, 200}]
```

Out[131]=



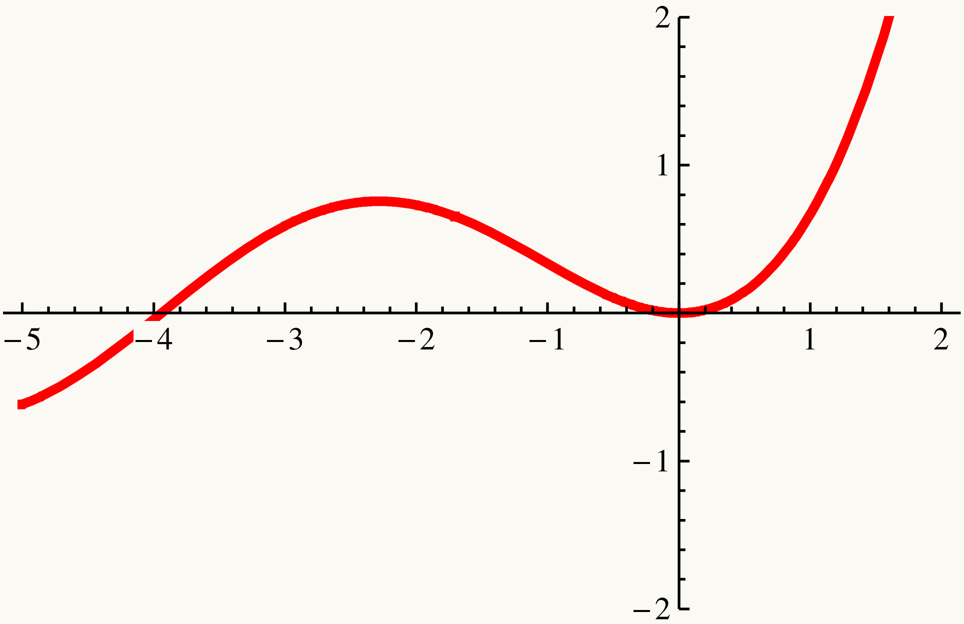
In[130]:=
```
sp = StreamPlot[{1, y + Sin[x]}, {x, -5, 2}, {y, -2, 2},
  ImageSize → {200, 200}]
```

Out[130]=



In[134]:=
```
traj1 = Plot[1 / 2 (Exp[x] - Sin[x] - Cos[x]), {x, -5, 2},
  PlotRange → {-2, 2}, PlotStyle → {Red, Thickness[.01]}]
```

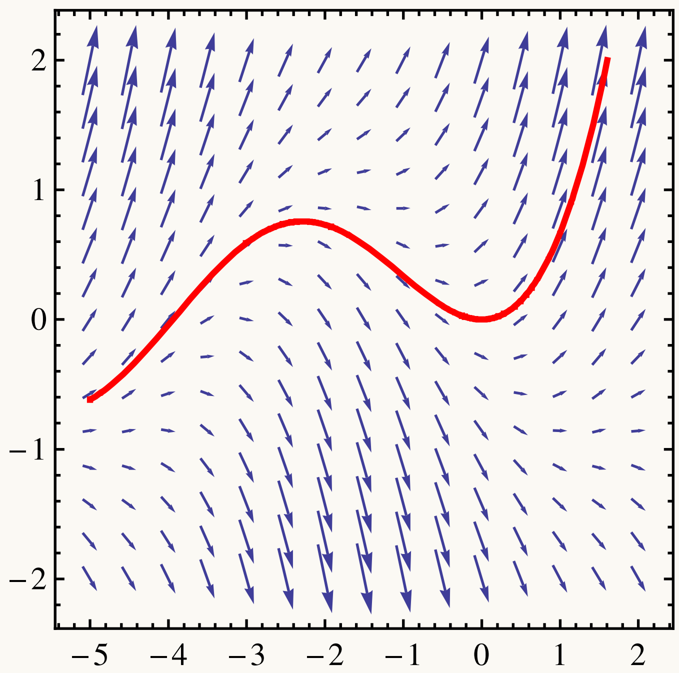
Out[134]=



In[138]:=

```
Show[vp, traj1, ImageSize → {250, 250}]
```

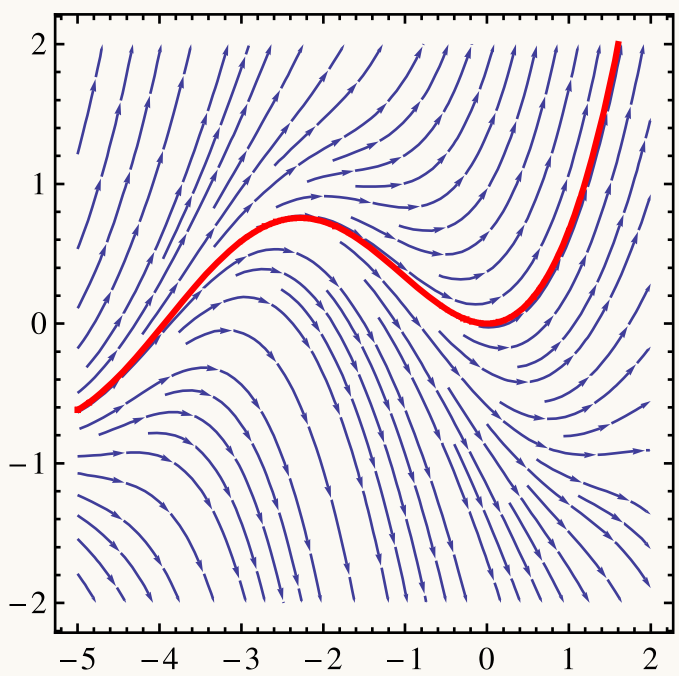
Out[138]=



In[137]:=

```
Show[sp, traj1, ImageSize → {250, 250}]
```

Out[137]=

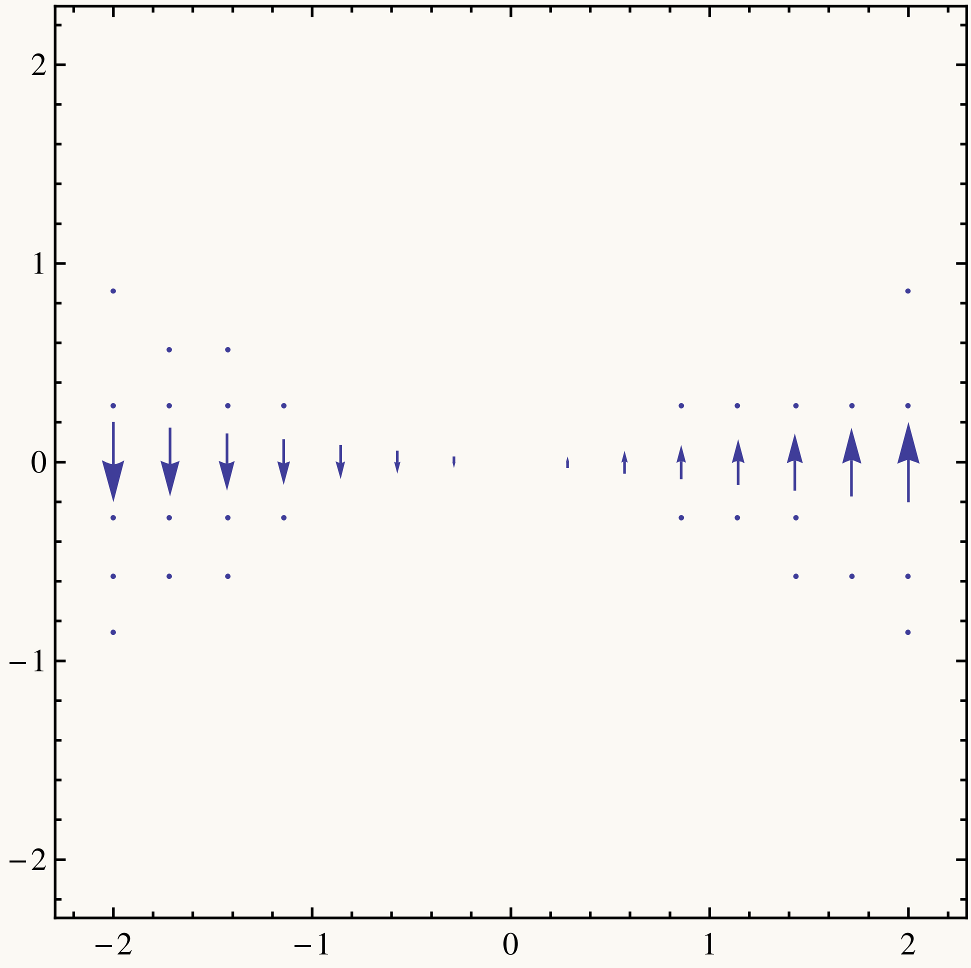


U některých i jednoduchých rovnic je nutné dávat pozor, jak se popíší souřadnice vektoru. Např. u následeující rovnice (řešení jsou kružnice) se při použití vektoru (1,-y/x) pro VectorPlot dostane nedobrý výsledek (u StreamPlot je obrázek celkem dobrý, kromě u osy x). Teprve při použití vektoru (y,-x) je i směrové pole v obou případech v pořádku.

```
Clear[y]; DSolve[ y'[x] y[x] + x == 0, y[x], x]
```

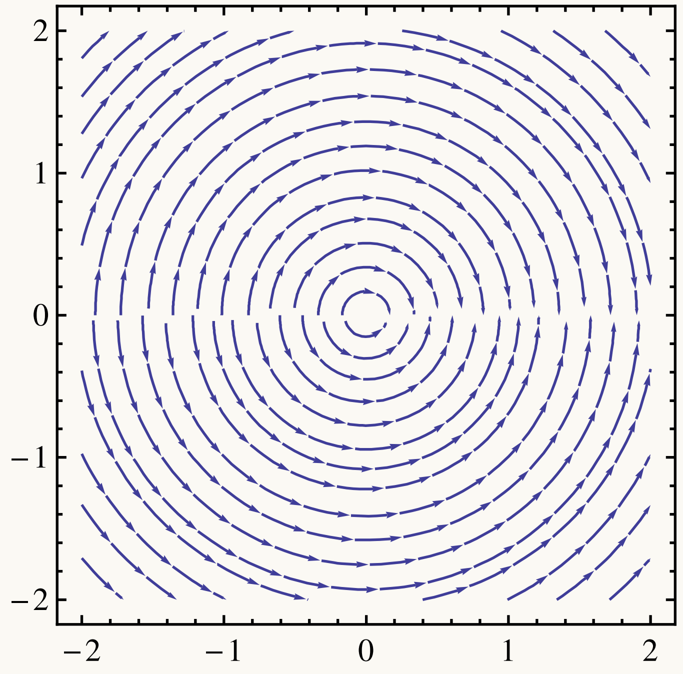
$$\left\{\left\{y(x)\to -\sqrt{2\,c_1 - x^2}\right\}, \left\{y(x)\to \sqrt{2\,c_1 - x^2}\right\}\right\}$$

```
VectorPlot[{1, -x / y}, {x, -2, 2}, {y, -2, 2}]
```



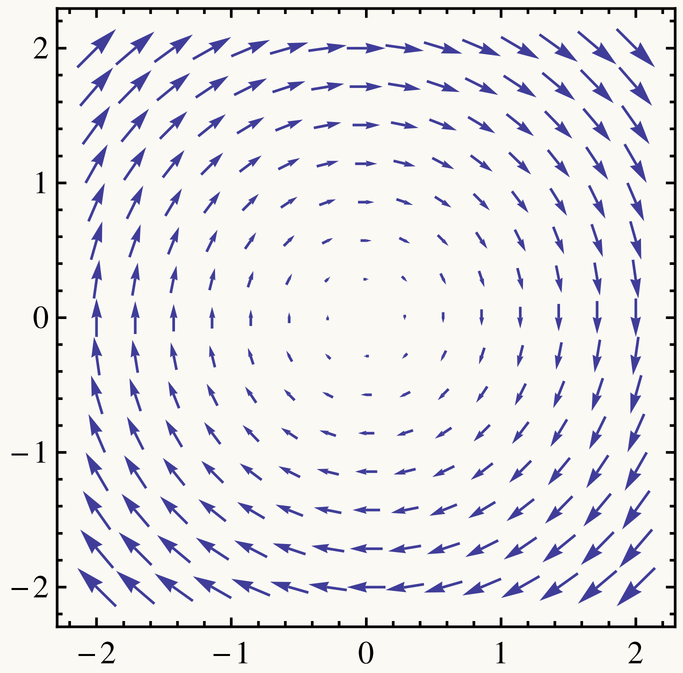
In[142]:=
```
StreamPlot[{1, -x / y}, {x, -2, 2}, {y, -2, 2},
 ImageSize → {250, 250}]
```

Out[142]=

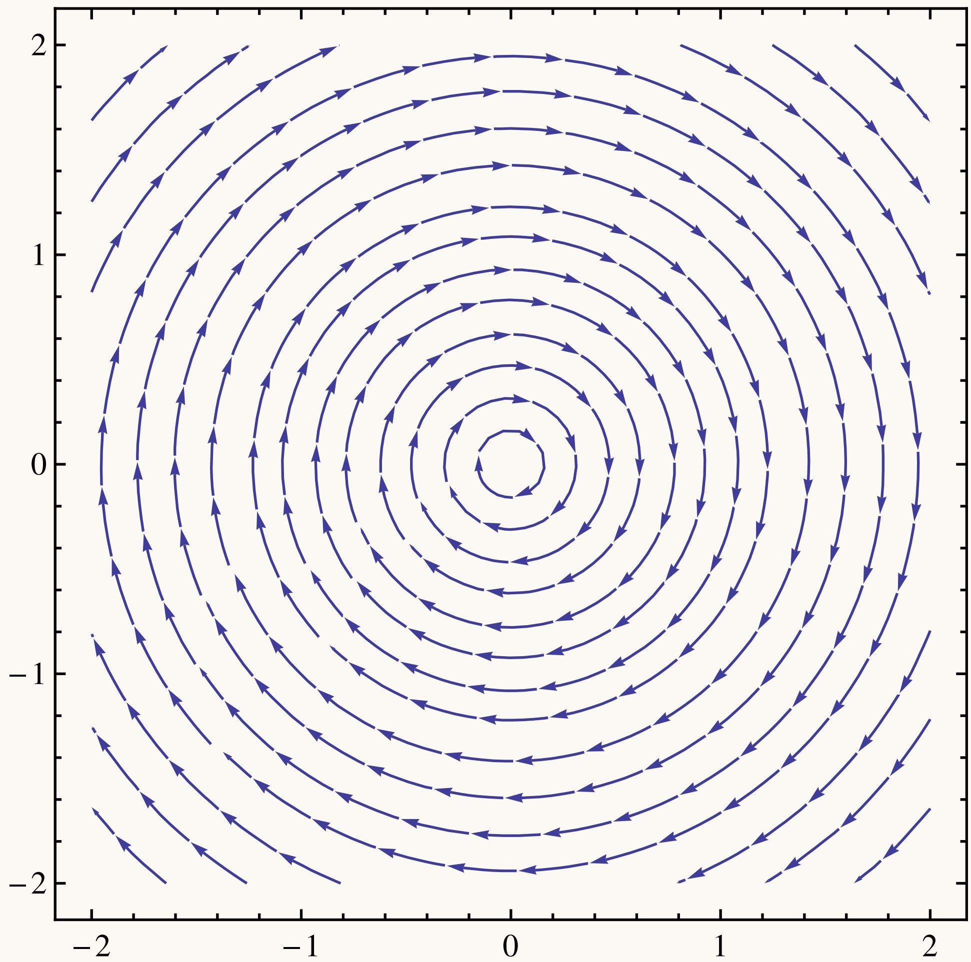


In[141]:=
```
VectorPlot[{y, -x}, {x, -2, 2}, {y, -2, 2},
 ImageSize → {250, 250}]
```

Out[141]=



```
StreamPlot[{y, -x}, {x, -2, 2}, {y, -2, 2}]
```



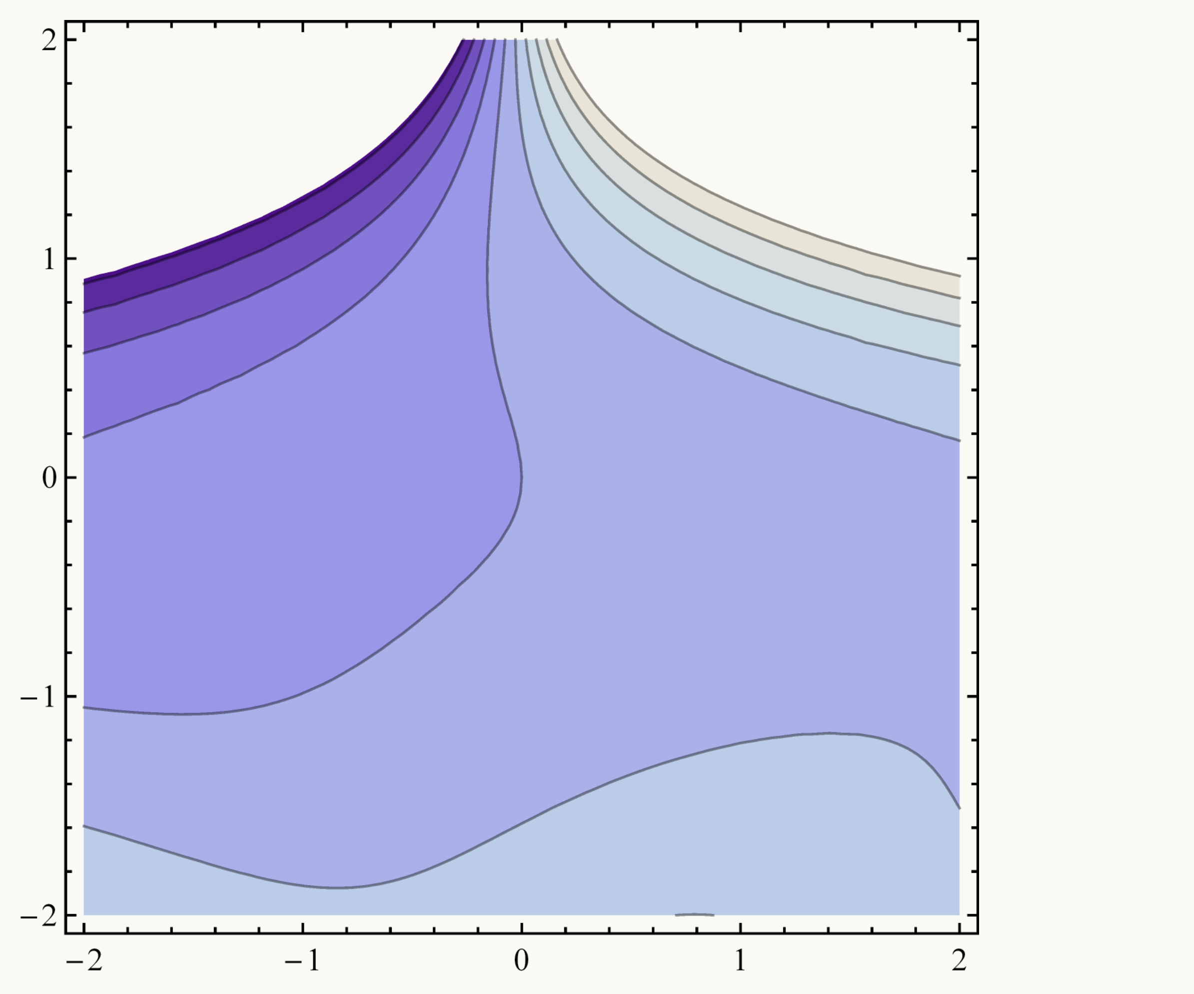
Mathematica dovede spočítat a ve vhodném tvaru rovnice se separovanými proměnnými, rovnice s homogenní funkcí, lineární rovnice a některá jejich zobecnění (např. Bernoulliovy nebo Riccatiho rovnice) a exaktní rovnice. V posledním případě se často dostane řešení v implicitním tvaru, což program vyznačí slovem „Solve" (viz následující příklad). Není třeba se obávat problémů u řešení uvedených rovnic kromě těch nejjednodušších, tj. se separovanými proměnnými (viz dále).

```
sol = DSolve[y'[x] (2 x Exp[2 y[x]] - x Cos[x y[x]] + 2 y[x]) ==
    y[x] Cos[x y[x]] - Exp[2 y[x]], y[x], x]
```

$\text{Solve}\left[y(x)^2 + x\, e^{2\, y(x)} - \sin(x\, y(x)) = c_1,\ y(x)\right]$

```
ContourPlot[Evaluate[sol[[1, 1]] /. {y[x] → y}],
 {x, -2, 2}, {y, -2, 2}]
```

— *Solve::nsmet : This system cannot be solved with the methods available to Solve.* ≫



Nyní se podíváme na jednoduchou rovnici y'=$\sqrt{y}$. Ta splňuje podmínky pro jednoznačnost řešení pro y>0. Přesto *Mathematica* dá dvě řešení procházející jedním bodem (1,1) - samozřejmě nemůže být jedno z nich řešením.

In[147]:=

```
Clear[y];
odrpart = DSolve[{y'[x] == Sqrt[y[x]], y[1] == 1}, y[x], x]
```

Out[147]=

$$\left\{\left\{y(x)\to\frac{1}{4}\left(x^2-6x+9\right)\right\},\left\{y(x)\to\frac{1}{4}\left(x^2+2x+1\right)\right\}\right\}$$

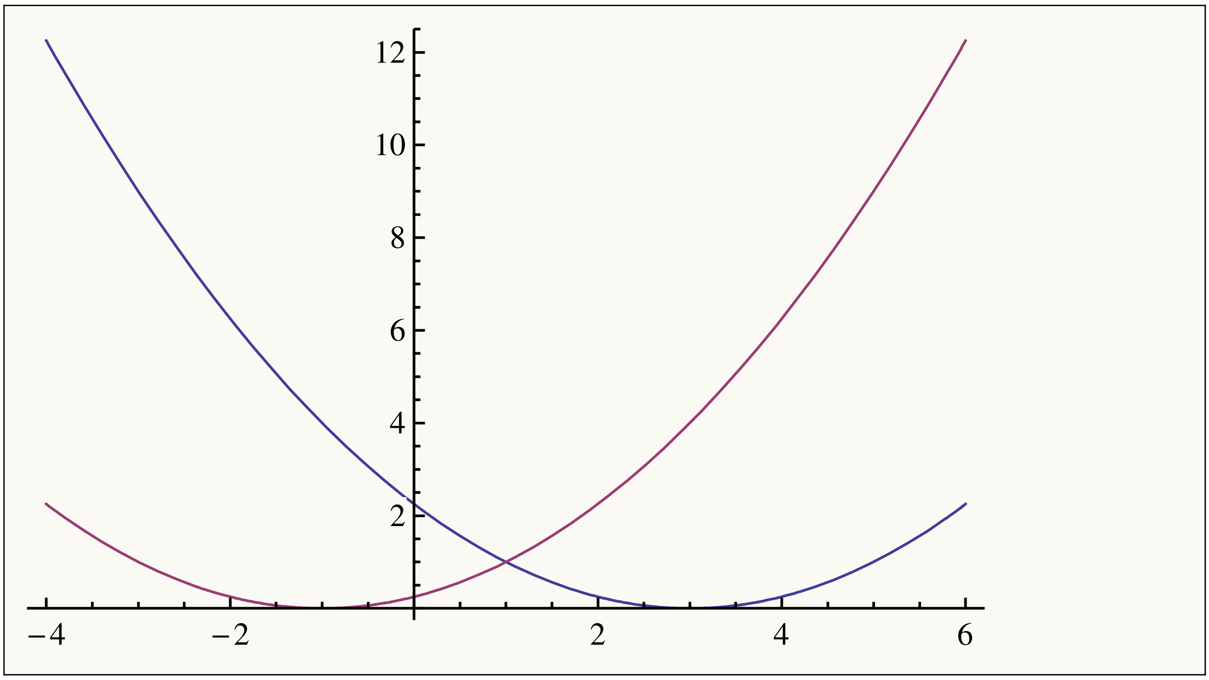
In[148]:=

```
reseni1 = odrpart[[1]]; reseni2 = odrpart[[2]];
y1[x_] = y[x] /. reseni1;
y2[x_] = y[x] /. reseni2;
```

In[149]:=

```
traj = Plot[{y1[x], y2[x]}, {x, -4, 6}]
```

Out[149]=

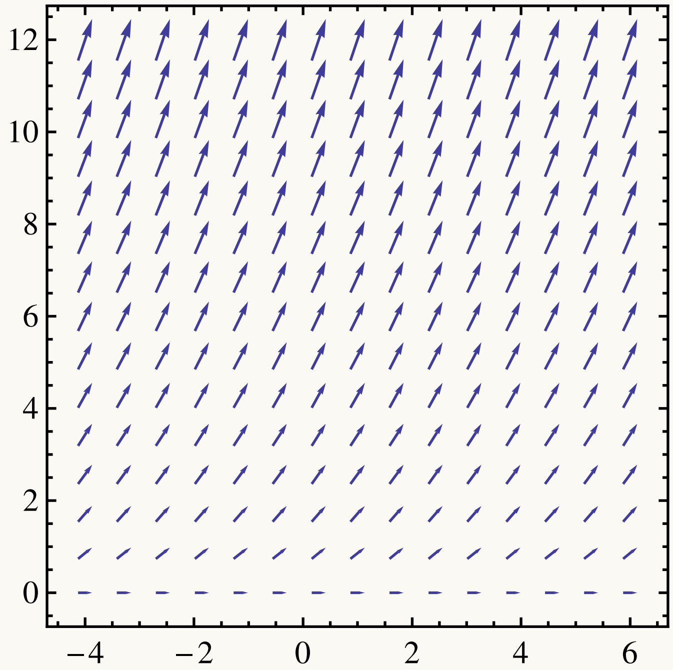


Nakreslíme směrové pole a do něj jedno řešení - z obrázku lze usoudit, kde je chyba.

In[143]:=

```
pole = VectorPlot[{1, Sqrt[y]}, {x, -4, 6}, {y, 0, 12},
  VectorScale → Small, ImageSize → {250, 250}]
```

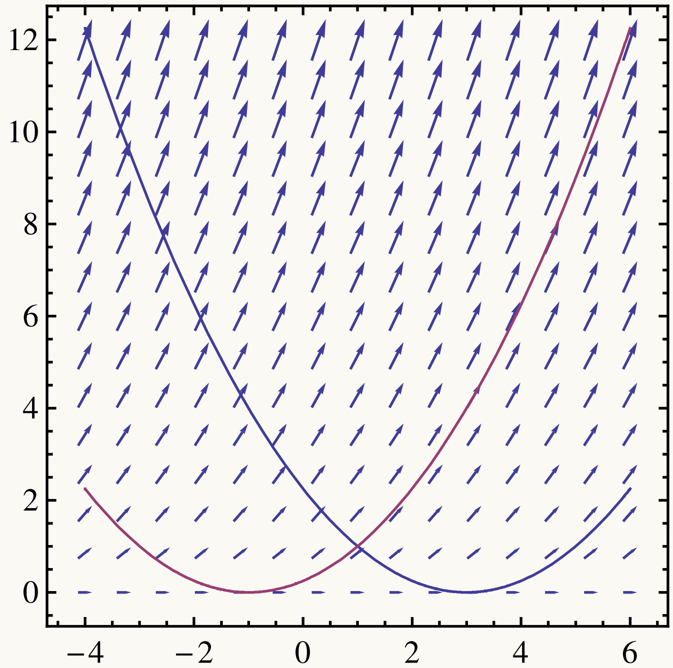
Out[143]=



In[150]:=

```
Show[pole, traj, ImageSize → {250, 250}]
```

Out[150]=



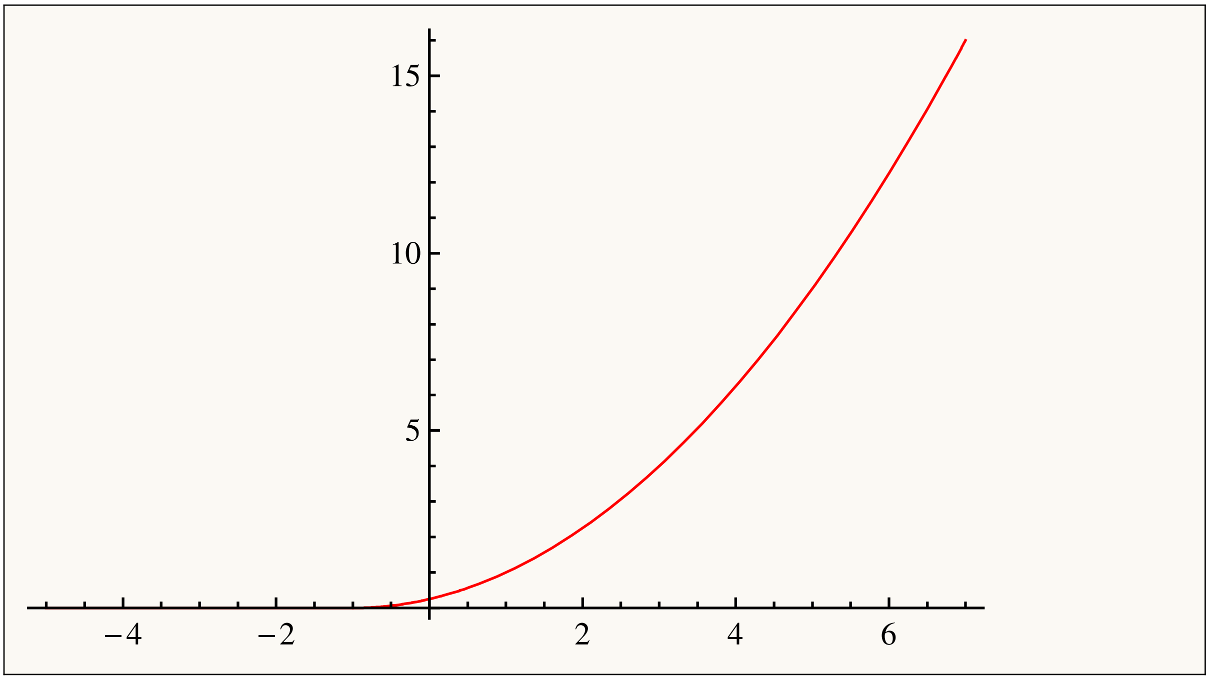
Mathematica (i podobné další programy) nedovede vést u všech zadání a kroků kontroly. Na to je třeba dávat pozor.

Správné řešení se získá kombinací nulové funkce a rostoucí části vypočteného „řešení".

In[154]:=
```
g[x_] := Piecewise[{{0, x < -1}, {y2[x], x ≥ -1}}]
```

In[160]:=
```
tra = Plot[g[x], {x, -5, 7}, PlotStyle → Red]
```

Out[160]=



In[161]:=
```
Show[pole, tra]
```

Out[161]=