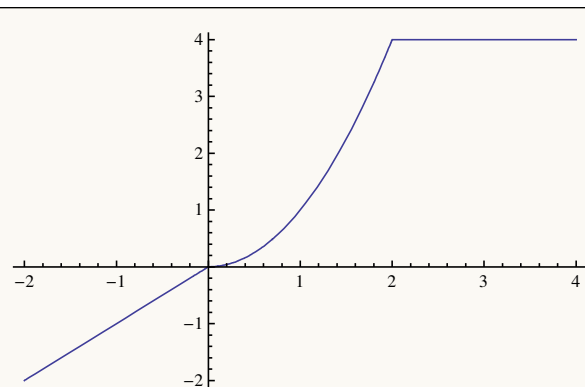


Funkce definovaná po částech

Občas je funkce definována různým vzorcem na různých intervalech, tj. po částech. V tomto případě existuje více možností zadání takové funkce, ta první pomocí `Piecewise` bývá nejhodnější. Ve druhém případě (definice pomocí `Which`) i třetím případě (definice pomocí podmínky), vyzkoušejte zadat stejné hodnoty jako v prvním případě.

```
f[x_] := Piecewise[{{x, x < 0}, {x^2, 0 < x < 2}}, 4]
```

```
Plot[f[x], {x, -2, 4}]
```



```
f[-5]
```

```
-5
```

```
f[1/2]
```

```
1/4
```

```
f[7]
```

```
4
```

```
f[2]
```

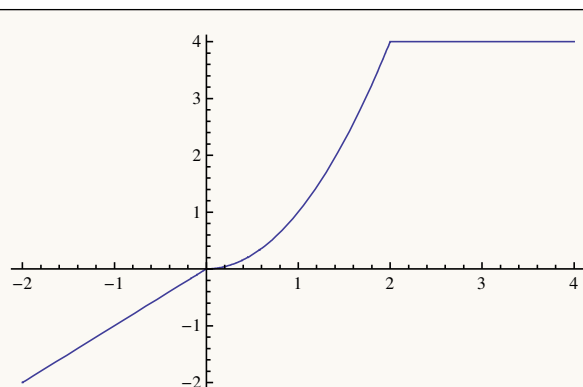
```
4
```

```
f[0]
```

```
4
```

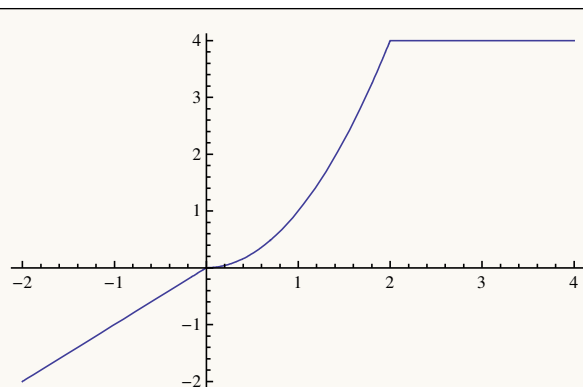
```
Clear[x, g]; g[x_] := x /; x ≤ 0; g[x_] := x^2 /; 0 < x < 2;  
g[x_] := 4 /; x ≥ 2
```

```
Plot[g[x], {x, -2, 4}]
```



```
Clear[x, h]; h[x_] := Which[x ≤ 0, x, 0 < x < 2, x^2, x ≥ 2, 4]
```

```
Plot[h[x], {x, -2, 4}]
```



Někdy bývá vhodné si připomenout definici po částech používané funkce. K tomu slouží příkaz [ExpandPiecewise](#), který pracuje pro funkce definované pomocí [Piecewise](#) a [While](#). Před použitím je nutné vyčistit nezávislou proměnnou.

```
Clear[x]; PiecewiseExpand[f[x]]
```

$$\begin{cases} 4 & x = 0 \vee x \geq 2 \\ x & x < 0 \\ x^2 & \text{True} \end{cases}$$

```
Clear[x]; PiecewiseExpand[g[x]]
```

$g(x)$

```
Clear[x]; PiecewiseExpand[h[x]]
```

$$\begin{cases} 4 & x \geq 2 \\ x & x \leq 0 \\ x^2 & \text{True} \end{cases}$$

Uvedený příkaz se hodí i v jiných situacích. Např. při definování funkce pomocí maxim nebo minim.

```
Clear[m]; m[x_] := Max[x^3, x];  
PiecewiseExpand[m[x_], 2 ≤ x ≤ 2]
```

$$\begin{cases} x_ & 0 \leq x_ \leq 1 \vee x_ \leq -1 \\ x_^3 & \text{True} \end{cases}$$

Nesouvislý definiční obor

Následující příklady uvádí problémy, s kterými se můžete setkat při zadávání funkce po částech pokud má funkce v bodě skok nebo není na nějakém "meziintervalu" definována.

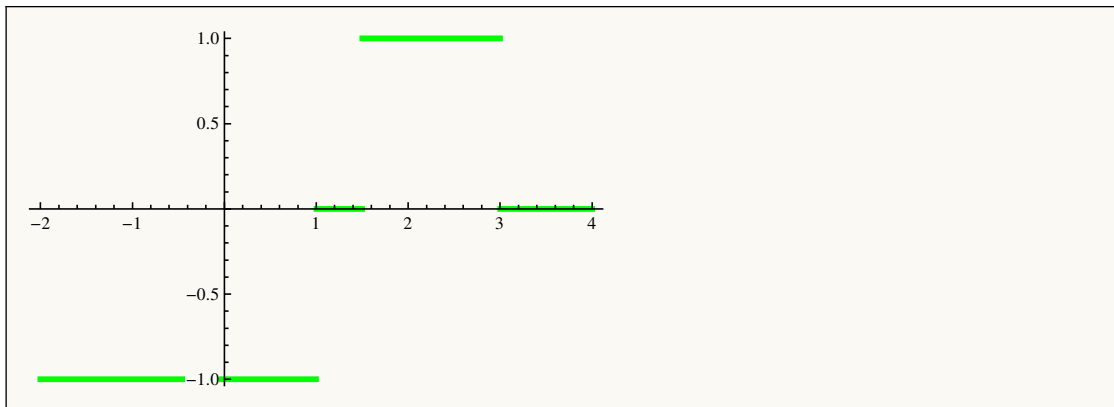
```
body = {0, 1, 1.2, 2, 4};
```

Piecewise dává většinou automaticky hodnotu 0 v bodech, kde funkce není definována.

```
Clear[x, f0];  
f0[x_] := Piecewise[{{-1, x < 1}, {1, 3 > x > 1.5}}];  
Map[f0, body]
```

{-1, 0, 0, 1, 0}

```
Plot[f0[x], {x, -2, 4}, PlotStyle -> {Green, Thickness[0.01]}
```



`Piecewise` má problémy se singulárními body. Podívejte se na následující dva příklady, ve kterých k předchozí funkci přidáme hodnotu v jednom bodě. Toto dodefinování program vyjádří jako hodnotu jiným způsobem (první řádka matice), má to však vliv na graf. V prvním případě dodáme hodnotu v bodě 1 v pořadí podle x - vše následující je ignorováno. Ve druhém případě dáme tuto hodnotu na konec definice -opět jsou další hodnoty ignorovány, ale graf je nyní bez dodání hodnot 0 tam, kde funkce není definována (hodnota v singulárním bodě však zakreslena není).

```
Clear[x, f1];
f1[x_] :=
  Piecewise[{{-1, x < 1}, {0.5, x = 1}, {1, 3 > x > 1.5}}];
Map[f1, body]
```

— *Set::setraw: Cannot assign to raw object 1. >>*

— *Set::setraw: Cannot assign to raw object 1.2. >>*

— *Set::setraw: Cannot assign to raw object 2. >>*

— *General::stop:*

Further output of Set::setraw will be suppressed during this calculation. >>

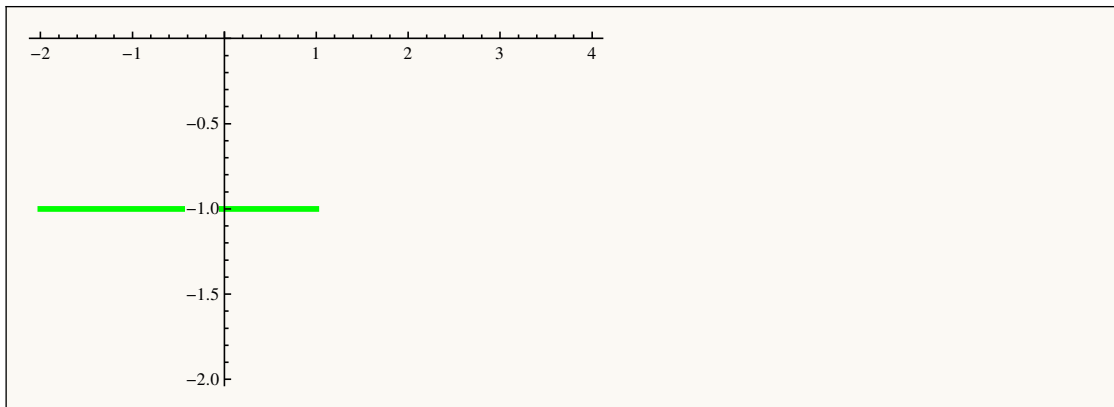
```
{-1, { 0.5 1
      0 True}, { 0.5 1
                0 True}, { 0.5 1
                          1 True}, { 0.5 1
                                      0 True}}
```

```
f1[1]
```

— *Set::setraw: Cannot assign to raw object 1. >>*

```
{ 0.5 1
  0 True}
```

```
Plot[f1[x], {x, -2, 4}, PlotStyle -> {Green, Thickness[0.01]}
```



```
Clear[x, f2];
f2[x_] :=
  Piecewise[{{-1, x < 1}, {1, 3 > x > 1.5}, {0.5, x = 1}}];
Map[f2, body]
```

— *Set::setraw: Cannot assign to raw object 1. >>*

— *Set::setraw: Cannot assign to raw object 1.2. >>*

— *Set::setraw: Cannot assign to raw object 4. >>*

— *General::stop:*

Further output of Set::setraw will be suppressed during this calculation. >>

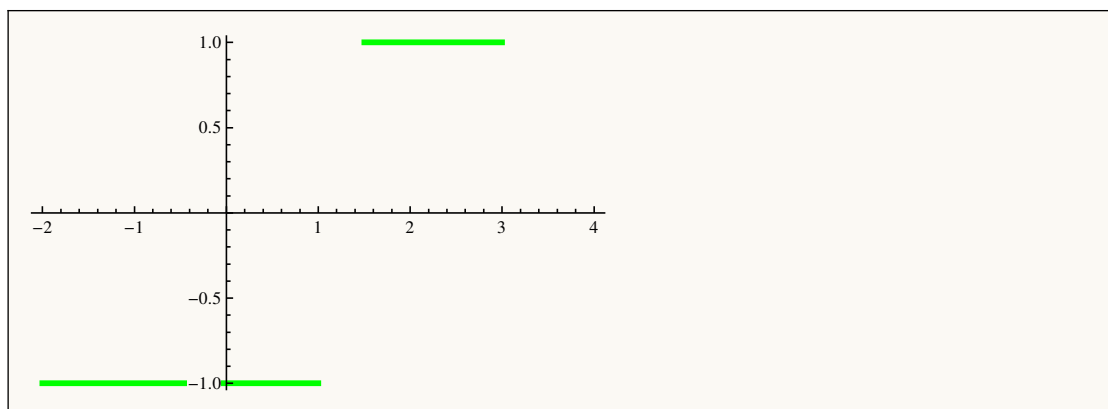
$$\{-1, \begin{Bmatrix} 0.5 & 1 \\ 0 & \text{True} \end{Bmatrix}, \begin{Bmatrix} 0.5 & 1 \\ 0 & \text{True} \end{Bmatrix}, 1, \begin{Bmatrix} 0.5 & 1 \\ 0 & \text{True} \end{Bmatrix}$$

```
f2[1]
```

— *Set::setraw: Cannot assign to raw object 1. >>*

$$\begin{Bmatrix} 0.5 & 1 \\ 0 & \text{True} \end{Bmatrix}$$

```
Plot[f2[x], {x, -2, 4}, PlotStyle -> {Green, Thickness[0.01]}
```

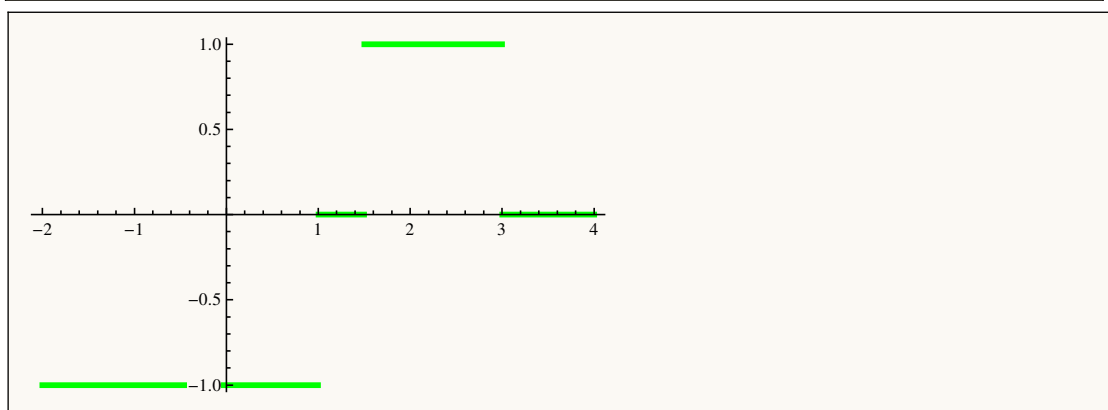


V těchto případech je vhodné dodefinovat funkci mimo `Piecewise`. Na graf to ale nemá vliv.

```
Clear[x, f3];
f3[x_] := Piecewise[{{-1, x < 1}, {1, 3 > x > 1.5}}];
f3[1] = 0.5;
Map[f3, body]
```

```
{-1, 0.5, 0, 1, 0}
```

```
Plot[f3[x], {x, -2, 4}, PlotStyle -> {Green, Thickness[0.01]}
```

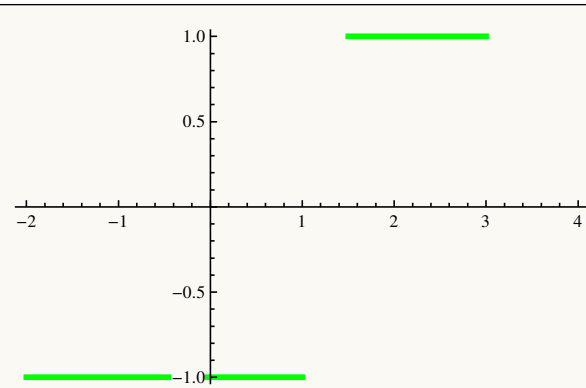


Použije-li se `Which`, je výsledek trochu jiný. V bodě, kde není funkce definována, je výsledkem `Null`. Ale singulární bod to pokazí i když tuto hodnotu dovede jednotlivě správně vypsát. (graf je správný, ale hodnota v singulárním bodě zobrazena není). Na rozdíl od `Piecewise` není funkce automaticky dodefinována.

```
Clear[x, g1]; g1[x_] := Which[x < 1, -1, 3 > x > 1.5, 1];
Map[g1, body]
```

```
{-1, Null, Null, 1, Null}
```

```
Plot[g1[x], {x, -2, 4}, PlotStyle -> {Green, Thickness[0.01]}]
```



```
Clear[x, g2];
g2[x_] := Which[x < 1, -1, 3 > x > 1.5, 1, 1 = x, 0.5];
Map[g2, body]
```

— *Set::setraw*: Cannot assign to raw object 1. >>

— *Set::setraw*: Cannot assign to raw object 1. >>

— *Set::setraw*: Cannot assign to raw object 1. >>

— *General::stop*:

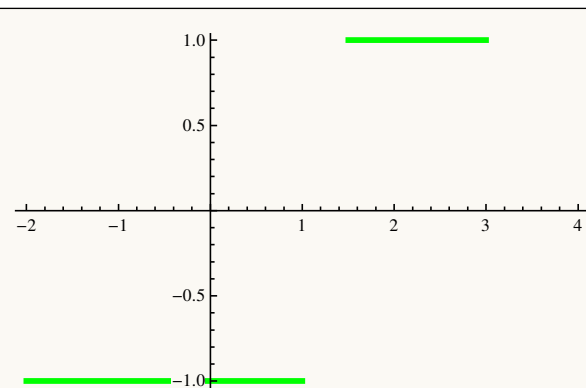
Further output of Set::setraw will be suppressed during this calculation. >>

```
{-1, Which[1, 0.5], Which[1.2, 0.5], 1, Which[4, 0.5]}
```

```
g3[1]
```

```
g3(1)
```

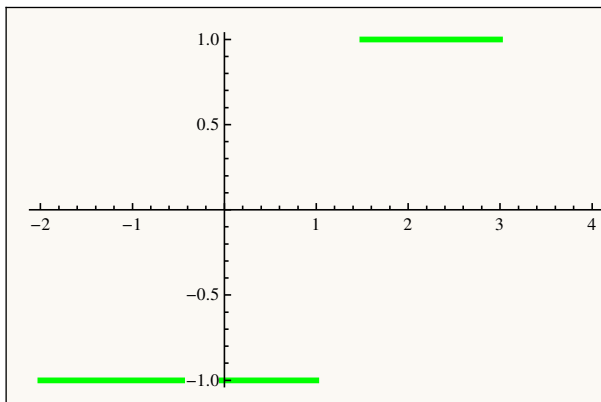
```
Plot[g2[x], {x, -2, 4}, PlotStyle -> {Green, Thickness[0.01]}]
```



```
Clear[x, g3]; g3[x_] := Which[x < 1, -1, 3 > x > 1.5, 1];
g3[1] = 0.5;
Map[g3, body]
```

```
{-1, 0.5, Null, 1, Null}
```

```
Plot[g3[x], {x, -2, 4}, PlotStyle -> {Green, Thickness[0.01]}]
```

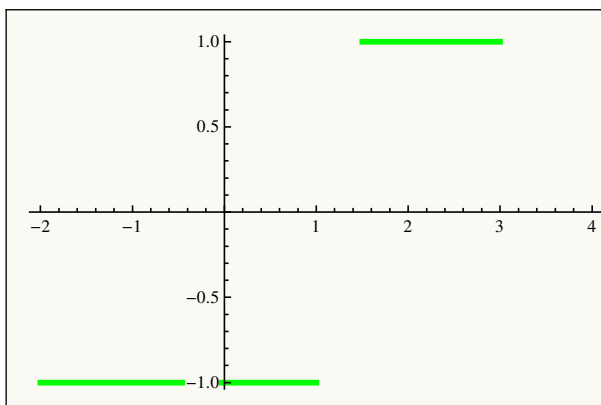


Při použití podmínky se chová funkce podobně jako při použití [While](#).

```
Clear[x, h1]; h1[x_] := -1 /; x < 1; h1[x_] := 1 /; 3 > x > 1.5;
Map[h1, body]
```

```
{-1, h1(1), h1(1.2), 1, h1(4)}
```

```
Plot[h1[x], {x, -2, 4}, PlotStyle -> {Green, Thickness[0.01]}]
```



```
Clear[x, h2]; h2[x_] := -1 /; x < 1; h2[x_] := 1 /; 3 > x > 1.5;
h2[1] = 0.5;
Map[h2, body]
```

```
{-1, 0.5, h2(1.2), 1, h2(4)}
```



```
Plot[h2[x], {x, -2, 4}, PlotStyle -> {Green, Thickness[0.01]}]
```

