Prepisovalna pravila

Mathematica je zasnovana kot **prepisovalni sistem**. To pomeni, da vsak izraz poenostavi tako, da ga **s prepisovalnimi pravili** predela iz začetne oblike v končno. Mnoga prepisovalna pravila so že vgrajena v Mathematico, denimo pravilo, ki prepiše izraz oblike

$$x + x + x$$

v izraz

$$3\,x$$

Poskusimo:

```
x + x + x
```

Tudi uporabnik lahko poda svoja prepisovalna pravila ter z njimi poenostavlja in obdeluje izraze. V tej lekciji se bomo naučili, kako uporabljamo prepisovalna pravila v Mathematici.

Prepisovalna pravila

Prepisovalno pravilo sestoji iz dveh delov, **vzorca** v in **izraza** e, zapišemo pa ga takole:

$$v \to e$$

Pomen takega pravila je "v se prepiše v e". Notacija za prepisovalna pravila je podobna matematični notaciji za preslikave,

$$x \mapsto f[x]$$

ki jo lahko preberemo tudi kot "x se prepiše v $f[x]$". Prepisovalna pravila so bolj splošna kot funkcije, ker lahko z njimi izrazimo tudi razna računaska pravila, kot je na primer "$x^2 - y^2$ se prepiše v $(x + y)\,(x - y)$":

$$x^2 - y^2 \;\to\; (x + y)\,(x - y)$$

- **Primeri prepisovalnih pravil:**

Konstanta `a` se prepiše v `1`:

```
a → 1
```

Izraz `f[-π]+f[π]` se prepiše v `0`:

```
(f[-π] + f[π]) → 0
```

Izraz $f[-x] + f[x]$ se prepiše v 0 za **poljuben** x:

```
(f[-x_] + f[x_]) → 0
```

Izraz oblike $\mathrm{Sin}[x + y]$ se prepiše v $\mathrm{Sin}[x]\,\mathrm{Cos}[y] + \mathrm{Cos}[x]\,\mathrm{Sin}[y]$ za **poljubna** x in y:

```
Sin[x_ + y_] → Sin[x] Cos[y] + Cos[x] Sin[y]
```

Zapomnimo si:

S **podčrtajem** označimo tiste spremenljivke v vzorcu, ki označujejo **poljuben** podizraz.

Kako uporabimo prepisovalno pravilo

Če je p prepisovalno pravilo in e izraz, potem lahko ***e* prepišemo s pravilom *p*** takole:

$$\mathrm{ReplaceAll}[e,\; p]$$

ali ekvivalentno

$$e\ /.\ p$$

Vsi podizrazi v e, ki ustrezajo vzorcu prepisovalnega pravila, se uzrezno prepišejo.

Primeri uporabe *e* /. *p*

V izrazu zamenjamo konstanto a z 1:

```
ReplaceAll[a^2 - 2 a b + b^2, a → 1]
```

```
(a^2 - 2 b + b^2) /. (a → 1)
```

Adicijski izrek za sinus lahko izrazimo kot prepisovalno pravilo:

```
adicijskiIzrek = (Sin[x_ + y_] → Sin[x] Cos[y] + Cos[x] Sin[y])
```

```
Sin[α + β + γ] /. adicijskiIzrek
```

Tole ne deluje po pričakovanjih:

```
Sin[α + α] /. adicijskiIzrek
```

Težava je v tem, da je Mathematica Sin[$\alpha + \alpha$] sama poenostavila v Sin[2 α], to pa je izraz, ki ne ustreza vzorcu Sin[x_ + y_].

- **Razlika med x in x_:**

Pomembno je, da razumemo, v čem je razlika med vzorcema `x` in `x_`. Prvi označuje **konstanto x**, drugi pa **poljuben izraz**. Na primer, če bi adicijski izrek zapisali takole

```
adicijskiIzrek2 = Sin[x + y] → Sin[x] Cos[y] + Cos[x] Sin[y]
```

potem bi zašli v težave:

```
Sin[a + b] /. adicijskiIzrek2
```

```
Sin[x + y] /. adicijskiIzrek2
```

Pravilo `adicijskiIzrek2` deluje samo na izrazu `Sin[x+y]`, kar je precej neuporabno.

Od posameznega primera je odvisno, ali želimo imeti v vzorcu konstano `x` ali poljuben `x_`. Na primer, če želimo v izrazu $1 + x + x^2$ zamenjati x s 3, potem to naredimo takole:

```
(1 + x + x^2) /. x → 3
```

Če bi uporabili prepisovalno pravilo `x_→3`, bi se celoten izraz prepisal v 3:

```
(1 + x + x^2) /. x_ → 3
```

Več prepisovalnih pravil hkrati

Pogosto želimo uporabiti več prepisovalnih pravil hkrati. Denimo, da želimo v izrazu $a^2 + a\,b + b^2$ zamenjati a z $1 + x$ in b z $1 - x$. To lahko naredimo z dvakratno uporabo /.:

```
(a^2 + a * b + b^2 /. a → 1 + x) /. b → 1 - x
```

Vendar je to dokaj nepraktično. Mathematica dopušča, da navedemo cel seznam pravil:

$$e\ /.\ \{p_1,\ \ldots,\ p_n\}$$

To pomeni "prepiši izraz e s prepisovalnimi pravili $p_1, \ldots, p_n$".

- **Primeri**

Zgornji primer lahko zapišemo bolj elegantno tako, da obe prepisovalni pravili združimo v seznam:

```
a^2 + a * b + b^2 /. {a → 1 + x, b → 1 - x}
```

Takole pa zamenjamo a in b:

```
Cos[a] + Sin[b] /. {a → b, b → a}
```

Kadar imamo več prepisovalnih pravil, je treba paziti na vrstni red. Mathematica uporabi prvo pravilo, ki se prilagaja izrazu.

```
f[x] + f[y] + f[z] /. {f[x] → 1, f[u_] → u}
```

```
f[x] + f[y] + f[z] /. {f[u_] → u, f[x] → 1}
```

Prepisovalna pravila in reševanje enačb s `Solve`

Ukaz **`Solve`** se v Mathematici uporablja za reševanje enačb. Rešitev enačbe ali sistema enačb je podana kot prepisovalno pravilo:

```
Solve[x^2 - x - 1 == 0, x]
```

Vsaka rešitev je podana kot seznam prepisovalnih pravil. **`Solve`** vrne seznam seznamov prepisovalnih pravil. Še en primer:

```
Solve[{x^2 + y^2 == 1, x^2 - x y - y^2 == 1}, {x, y}]
```

- **Naloga:** kolikšno vrednost ima izraz Sin[2 *x*], če je *x* rešitev enačbe Cos[*x*] + Sin[3 *x*] == 0?

Odgovor: ker **`Solve`** že poda rezultat kot prepisovalno pravilo, ga lahko uporabimo na izrazu Sin[2 *x*].

```
r = Sin[2 x] /. Solve[Cos[x] + Sin[3 x] == 0, x]
```

To lahko še poenostavimo s funkcijo **`FullSimplify`**:

```
FullSimplify[r]
```

Večkratno prepisovanje

Zapišimo še enkrat prepisovalno pravilo za adicijski izrek, tokrat za sin in cos:

```
adicijskiIzrek =
  { Sin[x_ + y_] → Sin[x] Cos[y] + Cos[x] Sin[y],
   Cos[x_ + y_] → Cos[x] Cos[y] - Sin[x] Sin[y]}
```

```
Sin[2 α + 3 β] + Cos[3 α + 2 β] /. adicijskiIzrek
```

Ko prepišemo s pravilom **`adicijskiIzrek`** izraz **`Sin[u+v+w]`**, se pravilo uporabi samo enkrat. Če ga želimo uporabiti dvakrat, moramo to storiti sami:

```
t = (Sin[u + v + w] /. adicijskiIzrek)
```

```
t /. adicijskiIzrek
```

Pogosto želimo prepisati izraz *e* s pravilom *p*, pri tem pa želimo prepisovanje ponavljati toliko časa, dokler se izraz spreminja. To naredimo z ukazom

$$\text{ReplaceRepeated}[e,\ p]$$

ali ekvivalentno

$$e\ //.\ p$$

To pomeni "prepisuj izraz *e* s pravilom *p*, dokler se spreminja"

- **Primeri uporabe *e* //. *p*:**

Enkrat uporabimo adicijski izrek z ukazom **`/.`**:

```
Sin[a + b + c + d + e] /. adicijskiIzrek
```

Če pa želimo uporabljati adicijske formule, dokler je to mogoče, uporabimo ukaz **`//.`**:

```
Sin[a + b + c + d + e] //. adicijskiIzrek
```

Kot drugi primer si oglejmo, kako lahko s prepisovalnimi pravili računamo člene rekurzivno definiranega zaporedja. Denimo, da so členi zaporedja $a[0]$, $a[1]$, … definirani s predpisom

$$a[0] == 1$$
$$a[1] == 1$$
$$a[n] == a[n-1] + a[n-2] \qquad (n\geq 2)$$

To zapišemo s prepisovalnim pravilom:

```
zaporedjePravilo = {
  a[0] → 1,
  a[1] → 1,
  a[n_] → a[n - 1] + a[n - 2]
 }
```

Vrednost člena izračunamo z ukazom `//.`:

```
a[3] //. zaporedjePravilo
```

```
a[15] //. zaporedjePravilo
```

```
{a[0], a[1], a[2], a[3], a[4], a[5], a[6], a[7]} //. zaporedjePravilo
```

Če pravilo uporabimo na splošnem členu, ga Mathematica prepisuje in prepisuje, dokler je ne prekinemo:

```
a[k] //. zaporedjePravilo
```

Težava je v tem, da tretje prepisovalno pravilo `a[n_]→a[n-1]+a[n-2]` ne izraža pogoja $n \geq 2$, zato ga Mathematica slepo uporablja:

```
a[k] /. zaporedjePravilo
```

```
a[k] /. zaporedjePravilo /. zaporedjePravilo
```

```
Simplify[a[k] /. zaporedjePravilo /. zaporedjePravilo /. zaporedjePravilo]
```

Če bi hoteli pravilno zapisati definicijo zaporedja $a[n]$, bi potrebovali tako imenovana **pogojna prepisovalna pravila**. Na tem mestu povejmo le, da zapišemo pogojno prepisovalno pravilo "vzorec v prepiši v izraz e, če je izpolnjen pogoj q" takole:

$$v\ /;\ q \to e$$

Primer:

```
Sin[α] Sin[0.0002] Cos[0.007] /. {Sin[x_] /; (x < 0.01) → x,
  Cos[x_] /; (x < 0.01) → (1 - x^2 / 2)}
```

Ko se bomo učili, kako se v Mathematici definira funkcije, bomo spoznali boljši način za delo z rekurzivno definiranimi zaporedji.

Povzetek

Prepisovalno pravilo "vzorec p se prepiše v izraz e" zapišemo v Mathematici $p \to e$

Poljuben podizraz v vzorcu označimo tako, da napišemo za ime spremenljivke podčrtaj.

Prepisovalno pravilo $p \to e$ uporabimo na izrazu a takole: a /. $p \to e$.
Več pravil uporabimo hkrati tako, da jih zberemo v seznam: a /. $\{p_1 \to e_1,\ p_2 \to e_2, \ldots.\ p_n \to e_n\}$

Funkcija `Solve` vrne rešitve sistema enačb kot seznam seznamov prepisovalnih pravil.
Vsaka rešitev je opisana z enim seznamom prepisovalnih števil.

Z ukazom a //. $p \to e$ izraz a prepisujemo s pravilom $p \to e$ toliko časa, dokler se spreminja.