2-AIN-266 Deklaratívne programovanie

Letný semester 2021/22

5. prednáška

Ján Komara

Obsah 5. prednášky

Distančná výučba

Pokyny

- Vypnite si kameru a stlmte si mikrofón.
- Komunikovať je možné počas celej prednášky.
- Ak máte otázku alebo chcete niečo povedať, tak zdvihnite ruku.
- Po vyzvaní zapnite si mikrofón.
- Po skončení ruku zložte a stlmte si mikrofón.
- Prezentácia je nahrávaná.
- Nahrávku z prednášky nájdete v tejto sekcii.
- Slajdy z prednášky nájdete pod položkou Súbory/Files.

Zopakovanie

Primitívne rekurzívne funkcie

- Základný vývoj primitívne rekurzívnych funkcií.
- Primitívne rekurzívne predikáty a ohraničená minimalizácia.
- Párovacia funkcia a aritmetizácia dátových štruktúr.
- Vnorená jednoduchá rekurzia:

$$
\begin{aligned}
f(0,y) &= g(y) \\
f(x+1,y) &= h\Big(x, f\left(x, s_1(x,y)\right), f\left(x, s_2(x,y,f(x,s_1(x,y)))\right), y\Big).
\end{aligned}
$$

- Regulárne rekurzívne definície s mierou:

$$f(\vec{x}) = \tau[f;\vec{x}].$$

  Podmienka regularity $\Gamma_{f(\vec{\rho})} \rightarrow \mu[\vec{\rho}] < \mu[\vec{x}]$ pre rekurzívne volanie $f(\vec{\rho})$ funkcie f v terme τ.

Zopakovanie

Vnorená jednoduchá rekurzia

Sú to definície v tvare

$$f(0,\vec{y}) = \rho[\vec{y}]$$
$$f(x+1,\vec{y}) = \tau[f(x,\cdot);x,\vec{y}].$$

Alternatívna formulácia (toto je klauzálna forma definície):

$$f(0,\vec{y}) = \rho[\vec{y}]$$
$$f(x+1,\vec{y}) = \theta[x,\vec{z},\vec{y}] \leftarrow \bigwedge_{i=1}^{k} f(x,\vec{\sigma}_i[x,\vec{y},z_1,\ldots,z_{i-1}]) = z_i.$$

Veta

Primitívne rekurzívne funkcie sú uzavreté na vnorenú jednoduchú rekurziu.

Príklady rekurzívnych definícii s mierou

Euklidov algoritmus pre výpočet najväčšieho spoločného deliteľa

Rekurzívna definícia s mierou $\max(x, y)$:

$$
\begin{aligned}
\gcd(x, y) = {} & \text{if } x \neq 0 \wedge y \neq 0 \text{ then} \\
& \quad \text{case} \\
& \quad\quad x > y \Rightarrow \gcd(x \dot{-} y, y) \\
& \quad\quad x = y \Rightarrow x \\
& \quad\quad x < y \Rightarrow \gcd(x, y \dot{-} x) \\
& \quad \text{end} \\
& \text{else} \\
& \quad \max(x, y).
\end{aligned}
$$

Podmienky regularity zaručujú, že výpočet vždy skončí:

$$
\begin{aligned}
x \neq 0 \wedge y \neq 0 \wedge x > y &\rightarrow \max(x \dot{-} y, y) < \max(x, y) \\
x \neq 0 \wedge y \neq 0 \wedge x < y &\rightarrow \max(x, y \dot{-} x) < \max(x, y).
\end{aligned}
$$

Príklady rekurzívnych definícii s mierou

Euklidov algoritmus pre výpočet najväčšieho spoločného deliteľa

Rekurzívna definícia s mierou $\max(x, y)$:

$$\begin{aligned}\gcd(x,y) = {} & \text{if } x \neq 0 \land y \neq 0 \text{ then} \\ & \quad \text{case} \\ & \qquad x > y \Rightarrow \gcd(x \dot{-} y, y) \\ & \qquad x = y \Rightarrow x \\ & \qquad x < y \Rightarrow \gcd(x, y \dot{-} x) \\ & \quad \text{end} \\ & \text{else} \\ & \quad \max(x,y).\end{aligned}$$

Podmienky regularity zaručujú, že výpočet vždy skončí:

$$x \neq 0 \land y \neq 0 \land x > y \rightarrow \max(x \dot{-} y, y) < \max(x, y)$$
$$x \neq 0 \land y \neq 0 \land x < y \rightarrow \max(x, y \dot{-} x) < \max(x, y).$$

Príklady rekurzívnych definícii s mierou

Aproximačná funkcia pre funkciu gcd

Špecifikácia aproximačnej funkcie

$$z > \max(x, y) \to f^+(z, x, y) = \gcd(x, y).$$

Definícia aproximačnej funkcie má tvar jednoduchej rekurzie

$$
\begin{aligned}
f^+(0, x, y) &= 0 \\
f^+(z+1, x, y) &= \text{if } x \neq 0 \land y \neq 0 \text{ then} \\
&\qquad \text{case} \\
&\qquad\quad x > y \Rightarrow f^+(z, x \dot{-} y, y) \\
&\qquad\quad x = y \Rightarrow x \\
&\qquad\quad x < y \Rightarrow f^+(z, x, y \dot{-} x) \\
&\qquad \text{end} \\
&\quad \text{else} \\
&\qquad \max(x, y).
\end{aligned}
$$

Primitívna rekurzívnosť $\gcd(x, y) = f^+\big(\max(x, y) + 1, x, y\big)$.

Príklady rekurzívnych definícii s mierou

Zarovnanie binárneho stromu

Klauzálna forma rekurzívnej definície:

$$
\begin{aligned}
\mathit{Flatten}(0) &= 0 \\
\mathit{Flatten}\,\langle u, 0\rangle &= \big\langle \mathit{Flatten}(u), 0\big\rangle \\
\mathit{Flatten}\,\big\langle u, \langle v, w\rangle\big\rangle &= \mathit{Flatten}\,\big\langle\langle u, v\rangle, w\big\rangle.
\end{aligned}
$$

Príklady rekurzívnych definícii s mierou

Zarovnanie binárneho stromu

Klauzálna forma rekurzívnej definície:

$$
\begin{aligned}
\mathit{Flatten}(0) &= 0 \\
\mathit{Flatten}\,\langle u, 0 \rangle &= \big\langle \mathit{Flatten}(u), 0 \big\rangle \\
\mathit{Flatten}\,\big\langle u, \langle v, w \rangle \big\rangle &= \mathit{Flatten}\,\big\langle \langle u, v \rangle, w \big\rangle.
\end{aligned}
$$

Príklady rekurzívnych definícii s mierou

Zarovnanie binárneho stromu

Klauzálna forma rekurzívnej definície:

$$\begin{aligned} Flatten(0) &= 0 \\ Flatten\,\langle u, 0\rangle &= \big\langle Flatten(u), 0\big\rangle \\ Flatten\,\big\langle u, \langle v, w\rangle\big\rangle &= Flatten\,\big\langle\langle u, v\rangle, w\big\rangle. \end{aligned}$$

Je to rekurzia s mierou $m(x)$:

$$\begin{aligned} m(0) &= 0 \\ m\,\langle v, w\rangle &= m(v) + 2m(w) + 1. \end{aligned}$$

Miera je primitívne rekurzívna funkcia (prečo?).
Podmienky regularity majú tvar

$$\begin{aligned} m(u) &< m\,\langle u, 0\rangle \\ m\,\big\langle\langle u, v\rangle, w\big\rangle &< m\,\big\langle u, \langle v, w\rangle\big\rangle. \end{aligned}$$

Príklady rekurzívnych definícii s mierou

Aproximačná funkcia pre funkciu *Flatten*

Špecifikácia aproximačnej funkcie

$$z > m(x) \to f^+(z, x) = \mathit{Flatten}(x).$$

Definícia aproximačnej funkcie má tvar jednoduchej rekurzie

$$\begin{aligned} f^+(0, x) &= 0 \\ f^+(z + 1, 0) &= 0 \\ f^+(z + 1, \langle u, 0\rangle) &= \big\langle f^+(z, u), 0\big\rangle \\ f^+\big(z + 1, \big\langle u, \langle v, w\rangle\big\rangle\big) &= f^+\big(z, \big\langle\langle u, v\rangle, w\big\rangle\big). \end{aligned}$$

Primitívna rekurzívnosť funkcie *Flatten* plynie z tohoto vyjadrenia

$$\mathit{Flatten}(x) = f^+(m(x) + 1, x).$$

Príklady rekurzívnych definícii s mierou

McCarthyho 91 funkcia

Definícia:

$$\mathrm{f}_{91}(x) = \begin{cases} 91 & \text{ak } x \leq 101, \\ x \dot{-} 10 & \text{ak } x \geq 101. \end{cases}$$

Alternatívne vyjadrenie pomocou vnorenej rekurzie

$$\mathrm{f}_{91}(x) = \text{if } x < 101 \text{ then } \mathrm{f}_{91}\,\mathrm{f}_{91}(x+11) \text{ else } x \dot{-} 10.$$

Príklady rekurzívnych definícii s mierou

McCarthyho 91 funkcia

Definícia:

$$\mathrm{f}_{91}(x) = \begin{cases} 91 & \text{ak } x \leq 101, \\ x \dot{-} 10 & \text{ak } x \geq 101. \end{cases}$$

Alternatívne vyjadrenie pomocou vnorenej rekurzie

$$\mathrm{f}_{91}(x) = \text{if } x < 101 \text{ then } \mathrm{f}_{91}\, \mathrm{f}_{91}(x + 11) \text{ else } x \dot{-} 10.$$

Príklady rekurzívnych definícii s mierou

McCarthyho 91 funkcia

Definícia:

$$\mathrm{f}_{91}(x) = \begin{cases} 91 & \text{ak } x \leq 101, \\ x \dot{-} 10 & \text{ak } x \geq 101. \end{cases}$$

Alternatívne vyjadrenie pomocou vnorenej rekurzie

$$\mathrm{f}_{91}(x) = \text{if } x < 101 \text{ then } \mathrm{f}_{91}\,\mathrm{f}_{91}(x+11) \text{ else } x \dot{-} 10.$$

Je to korektná definícia?
Tvrdíme, že je to spätná rekurzia s mierou $101 \dot{-} x$. Podmienky regularity majú teda tvar

$$x < 101 \rightarrow 101 \dot{-} (x+11) < 101 \dot{-} x$$
$$x < 101 \rightarrow 101 \dot{-} \mathrm{f}_{91}(x+11) < 101 \dot{-} x.$$

Ako splniť druhú podmienku?

Príklady rekurzívnych definícii s mierou

McCarthyho 91 funkcia

Definícia:

$$\mathrm{f}_{91}(x) = \begin{cases} 91 & \text{ak } x \leq 101, \\ x \dot{-} 10 & \text{ak } x \geq 101. \end{cases}$$

Alternatívne vyjadrenie pomocou vnorenej rekurzie

$$\mathrm{f}_{91}(x) = \text{if } x < 101 \text{ then } \mathrm{f}_{91}\,\mathrm{f}_{91}(x+11) \text{ else } x \dot{-} 10.$$

Je to korektná definícia?
Tvrdíme, že je to spätná rekurzia s mierou $101 \dot{-} x$. Podmienky regularity majú teda tvar

$$x < 101 \rightarrow 101 \dot{-} (x+11) < 101 \dot{-} x$$
$$x < 101 \rightarrow 101 \dot{-} \mathrm{f}_{91}(x+11) < 101 \dot{-} x.$$

Ako splniť druhú podmienku?

Príklady rekurzívnych definícii s mierou

McCarthyho 91 funkcia

Definícia:

$$\mathrm{f}_{91}(x) = \begin{cases} 91 & \text{ak } x \leq 101, \\ x \dot{-} 10 & \text{ak } x \geq 101. \end{cases}$$

Alternatívne vyjadrenie pomocou vnorenej rekurzie

$$\mathrm{f}_{91}(x) = \text{if } x < 101 \text{ then } \mathrm{f}_{91}\, \mathrm{f}_{91}(x+11) \text{ else } x \dot{-} 10.$$

Je to korektná definícia?
Tvrdíme, že je to spätná rekurzia s mierou $101 \dot{-} x$. Podmienky regularity majú teda tvar

$$x < 101 \rightarrow 101 \dot{-} (x+11) < 101 \dot{-} x$$
$$x < 101 \rightarrow 101 \dot{-} \mathrm{f}_{91}(x+11) < 101 \dot{-} x.$$

Ako splniť druhú podmienku?

Príklady rekurzívnych definícii s mierou

Splnenie podmienok regularity pre McCarthyho 91 funkciu

Uvažujme funkciu f definovanou rekurzívnou rovnosťou

$$f(x) = \text{if } x < 101 \text{ then } [f]_x[f]_x(x+11) \text{ else } x \dot{-} 10. \qquad (1)$$

Tu $[f]_x(y)$ je *zúženie* f na vstupy y také, že $101 \dot{-} y < 101 \dot{-} x$:

$$[f]_x(y) \equiv \text{if } 101 \dot{-} y < 101 \dot{-} x \text{ then } f(y) \text{ else } 0. \qquad (2)$$

Každá rekurzívna aplikácia v (1) je strážená kontextom v tvare (2).
Je to korektná definícia.
Funkcia f spĺňa podmienky regularity pôvodnej rekurzívnej rovnice:

$$x < 101 \to 101 \dot{-} (x+11) < 101 \dot{-} x$$
$$x < 101 \to 101 \dot{-} f(x+11) < 101 \dot{-} x.$$

Táto funkcia je tiež jediným riešením pôvodnej rekurzívnej rovnice.

Príklady rekurzívnych definícii s mierou

McCarthyho 91 funkcia

Definícia:

$$\mathrm{f}_{91}(x) = \begin{cases} 91 & \text{ak } x \leq 101, \\ x \dot{-} 10 & \text{ak } x \geq 101. \end{cases}$$

Alternatívne vyjadrenie pomocou vnorenej rekurzie

$$\mathrm{f}_{91}(x) = \text{if } x < 101 \text{ then } \mathrm{f}_{91}\,\mathrm{f}_{91}(x + 11) \text{ else } x \dot{-} 10.$$

Je to spätná rekurzia s mierou $101 \dot{-} x$. Podmienky regularity majú teda tvar

$$x < 101 \to 101 \dot{-} (x + 11) < 101 \dot{-} x$$
$$x < 101 \to 101 \dot{-} \mathrm{f}_{91}(x + 11) < 101 \dot{-} x.$$

Príklady rekurzívnych definícii s mierou

Aproximačná funkcia pre McCarthyho 91 funkciu

Špecifikácia aproximačnej funkcie

$$z > 101 \dot{-} x \rightarrow f^+(z, x) = \mathrm{f}_{91}(x).$$

Definícia aproximačnej funkcie má tvar vnorenej jednoduchej rekurzie

$$\begin{aligned}
f^+(0, x) &= 0 \\
f^+(z+1, x) &= \text{if } x < 101 \text{ then} \\
&\qquad f^+\big(z, f^+(z, x+11)\big) \\
&\quad \text{else} \\
&\qquad x \dot{-} 10.
\end{aligned}$$

Primitívna rekurzívnosť McCarthyho 91 funkcie plynie z tohoto vyjadrenia

$$\mathrm{f}_{91}(x) = f^+(101 \dot{-} x + 1, x).$$

Syntaktická forma rekurzie s mierou

Rekurzia s mierou $\mu[\vec{x}]$

Sú to definície v tvare

$$f(\vec{x}) = \tau\big[[f]^{\mu}_{\vec{x}}; \vec{x}\big]. \tag{1}$$

Term na pravej strane rovnosti vznikol s termu $\tau[f; \vec{x}]$ nahradením každej rekurzívnej aplikácie $f(\vec{\rho})$ výrazom

$$[f]_{\vec{x}}(\vec{\rho}) \equiv \text{if } \mu[\vec{\rho}] < \mu[\vec{x}] \text{ then } f(\vec{\rho}) \text{ else } 0.$$

Každá rekurzívna aplikácia $f(\vec{\rho})$ v (1) je teda strážená podmienkou $\mu[\vec{\rho}] < \mu[\vec{x}]$.

Veta

Primitívne rekurzívne funkcie sú uzavreté na rekurziu s mierou.

Syntaktická forma rekurzie s mierou

Dôkaz

Špecifikácia aproximačnej funkcie

$$z > \mu[\vec{x}] \rightarrow f^+(z, \vec{x}) = f(\vec{x}).$$

Definícia aprox. funkcie má tvar vnorenej jednoduchej rekurzie

$$\begin{aligned} f^+(0, \vec{x}) &= 0 \\ f^+(z+1, \vec{x}) &= \tau\big[[f^+]^{\mu}_{z,\vec{x}}; z, \vec{x}\big]. \end{aligned}$$

Term na pravej strane druhej rovnosti vznikol s termu $\tau[f; \vec{x}]$ nahradením každej rekurzívnej aplikácie $f(\vec{\rho})$ výrazom

$$[f^+]^{\mu}_{z,\vec{x}}(\vec{\rho}) \equiv \text{if } \mu[\vec{\rho}] < \mu[\vec{x}] \text{ then } f^+(z, \vec{\rho}) \text{ else } 0.$$

Primitívna rekurzívnosť funkcie f plynie z tohoto vyjadrenia

$$f(\vec{x}) = f^+(\mu[\vec{x}] + 1, \vec{x}).$$

Regulárne rekurzívne definície s mierou

Regulárna rekurzia s mierou $\mu[\vec{x}]$

Sú to definície v tvare

$$f(\vec{x}) = \tau[f;\vec{x}], \tag{1}$$

pre ktoré sú splnené podmienky regularity.
Podmienky regularity majú tvar:

$$\Gamma^{\tau}_{f(\vec{\rho})}[f;\vec{x}] \to \mu[\vec{\rho}[f;\vec{x}]] < \mu[\vec{x}]$$

pre každú rekurzívnu aplikáciu $f(\vec{\rho})$ funkcie f, ktorá je strážená podmienkou $\Gamma^{\tau}_{f(\vec{\rho})}$ v terme τ.
Splnenie podmienok regularity sa overuje pre funkciu, ktorá je definovaná pridruženou rovnosťou:

$$f(\vec{x}) = \tau\big[[f]^{\mu}_{\vec{x}};\vec{x}\big].$$

Táto funkcia je potom jediným riešením funkcionálnej rovnice (1).

Regulárne rekurzívne definície s mierou

Veta

Primitívne rekurzívne funkcie sú uzavreté na regulárnu rekurziu s mierou.

Dôkaz

Pridružená rovnosť má tvar syntaktickej formy rekurzie s mierou:

$$f(\vec{x}) = \tau\left[[f]^{\mu}_{\vec{x}}; \vec{x}\right].$$

Funkcia f definovaná týmto vzťahom je preto primitívne rekurzívna.

Poznámka

Číslo $\mu[\vec{x}] + 1$ predstavuje horný odhad na hĺbku výpočtového stromu pre výpočet aplikácie $f(\vec{x})$ pomocou programu

$$f(\vec{x}) = \tau[f; \vec{x}].$$

Charakterizačný problém

Veta

Trieda primitívne rekurzívnych funkcií je najmenšia trieda funkcií, ktorá obsahuje funkciu nasledovníka $\mathrm{S}(x) = x + 1$, *funkciu predchodcu* $x \dot{-} 1$ *a je uzavretá na explicitné definície a regulárnu rekurziu s mierou.*

Dôkaz.

Označenie:

- PRIM = trieda primitívne rekurzívnych funkcií.
- REG = najmenšia trieda funkcií, ktorá obsahuje funkciu nasledovníka $\mathrm{S}(x) = x + 1$, funkciu predchodcu $x \dot{-} 1$ a je uzavretá na explicitné definície a regulárnu rekurziu s mierou.

Cieľom je dokázať rovnosť

$$\mathrm{PRIM} = \mathrm{REG}.$$

Charakterizačný problém

Dôkaz inklúzie PRIM $\subseteq$ REG

- S $\in$ REG, pretože S je základná funkcia triedy REG.
- Z $\in$ REG plynie z tohoto vyjadrenia $\mathrm{Z}(x) = 0$.
- $\mathrm{I}_i^n \in$ REG plynie z tohoto vyjadrenia $\mathrm{I}_i^n(x_1, \ldots, x_n) = x_i$.
- Trieda REG je uzavretá na operátor kompozície funkcií:

$$f(x_1, \ldots, x_n) = h\big(g_1(x_1, \ldots, x_n), \ldots, g_m(x_1, \ldots, x_n)\big).$$

  Je to totiž špeciálny prípad explicitnej definície.
- Trieda REG je uzavretá na operátor primitívnej rekurzie:

$$f(0, \vec{y}) = g(\vec{y}) \qquad f(x+1, \vec{y}) = h\big(x, f(x, \vec{y}), \vec{y}\big).$$

  Je to špeciálny prípad regulárnej rekurzie s mierou $\mu[x, \vec{y}] \equiv x$:

$$f(x, \vec{y}) = \text{if } x \neq 0 \text{ then } h\big(x \dot{-} 1, f(x \dot{-} 1, \vec{y}), \vec{y}\big) \text{ else } g(\vec{y}).$$

Charakterizačný problém

Veta

Trieda primitívne rekurzívnych funkcií je najmenšia trieda funkcií, ktorá obsahuje funkciu nasledovníka $\mathrm{S}(x) = x + 1$*, funkciu predchodcu* $x \dot{-} 1$ *a je uzavretá na explicitné definície a regulárnu rekurziu s mierou.*

Dôkaz.

Označenie:

- PRIM = trieda primitívne rekurzívnych funkcií.
- REG = najmenšia trieda funkcií, ktorá obsahuje funkciu nasledovníka $\mathrm{S}(x) = x + 1$, funkciu predchodcu $x \dot{-} 1$ a je uzavretá na explicitné definície a regulárnu rekurziu s mierou.

Cieľom je dokázať rovnosť

$$\mathrm{PRIM} = \mathrm{REG}.$$

Charakterizačný problém

Dôkaz inklúzie REG $\subseteq$ PRIM

- S $\in$ PRIM, pretože S je základná funkcia triedy PRIM.
- $\lambda x.x \dot{-} 1 \in$ PRIM plynie z tohoto vyjadrenia

$$0 \dot{-} 1 = 0 \qquad\qquad x + 1 \dot{-} 1 = x.$$

- Trieda PRIM je uzavretá na explicitné definície funskcií:

$$f(x_1, \ldots, x_n) = \tau[x_1, \ldots, x_n].$$

  Dôkaz tvrdenia na 2. prednáške.
- Trieda PRIM je uzavretá na regulárnu rekurziu s mierou:

$$f(x_1, \ldots, x_n) = \tau[f; x_1, \ldots, x_n].$$

  Dôkaz tvrdenia na tejto prednáške.

Deklaratívne programovanie

Programy deklaratívnej paradigmy

Regulárne rekurzívne definície s mierou:

$$f(\vec{x}) = \tau[f; \vec{x}].$$

Základné konštrukcie pri tvorbe programov

- Premenné a konštanty.
- Funkčné aplikácie.
- Podmienkové výrazy.
  - Jednoduché podmienkové výrazy:

$$\mathrm{D}(\tau_1, \tau_2, \tau_3) \equiv \text{if } \tau_1 \neq 0 \text{ then } \tau_2 \text{ else } \tau_3.$$

  - Obecné podmienkové výrazy:

$$\text{case } \varphi_1 \Rightarrow \rho_1 \ \ldots \ \varphi_m \Rightarrow \rho_m \text{ end.}$$

Deklaratívne programovanie

Diskriminácia na konštantách

Rekurzívna definícia Fibonacciho postupnosti:

$$
\begin{aligned}
f_n = \mathsf{case} \\
&\quad n = 0 \Rightarrow 0 \\
&\quad n = 1 \Rightarrow 1 \\
&\quad n \neq 0 \wedge n \neq 1 \Rightarrow f_{n \dot{-} 1} + f_{n \dot{-} 2} \\
&\mathsf{end}.
\end{aligned}
$$

Program:

$$
\begin{aligned}
f_n = \mathsf{case} \\
&\quad (n =_* 0) = 1 \Rightarrow 0 \\
&\quad (n =_* 1) = 1 \Rightarrow 1 \\
&\quad (n \neq_* 0 \wedge_* n \neq_* 1) = 1 \Rightarrow f_{n \dot{-} 1} + f_{n \dot{-} 2} \\
&\mathsf{end}.
\end{aligned}
$$

Preklad do definície s jednoduchými podmienkovými výrazmi:

$$f_n = \mathsf{if}\ (n =_* 0) \neq 0\ \mathsf{then}\ 0\ \mathsf{else}\ \mathsf{if}\ (n =_* 1) \neq 0\ \mathsf{then}\ 1\ \mathsf{else}\ f_{n \dot{-} 1} + f_{n \dot{-} 2}.$$

Deklaratívne programovanie

Diskriminácia na konštantách

Rekurzívna definícia Fibonacciho postupnosti:

$$\begin{aligned} f_n = \text{case} \\ & n = 0 \Rightarrow 0 \\ & n = 1 \Rightarrow 1 \\ & \text{otherwise} \Rightarrow f_{n \dot{-} 1} + f_{n \dot{-} 2} \\ \text{end}. \end{aligned}$$

Program:

$$\begin{aligned} f_n = \text{case} \\ & (n =_* 0) = 1 \Rightarrow 0 \\ & (n =_* 1) = 1 \Rightarrow 1 \\ & (n \geq_* 2) = 1 \Rightarrow f_{n \dot{-} 1} + f_{n \dot{-} 2} \\ \text{end}. \end{aligned}$$

Preklad do definície s jednoduchými podmienkovými výrazmi:

$$f_n = \text{if } (n =_* 0) \neq 0 \text{ then } 0 \text{ else if } (n =_* 1) \neq 0 \text{ then } 1 \text{ else } f_{n \dot{-} 1} + f_{n \dot{-} 2}.$$

Dichotomická diskriminácia

Explicitná definícia funkcie $\max(x, y)$:

$$\begin{aligned} \max(x, y) = \text{case} & \\ & x \geq y \Rightarrow x \\ & x < y \Rightarrow y \\ \text{end.} & \end{aligned}$$

Program:

$$\begin{aligned} \max(x, y) = \text{case} & \\ & (x \geq_* y) = 1 \Rightarrow x \\ & (x <_* y) = 1 \Rightarrow y \\ \text{end.} & \end{aligned}$$

Preklad do definície s jednoduchým podmienkovým výrazom:

$$\max(x, y) = \text{if } (x \geq_* y) \neq 0 \text{ then } x \text{ else } y.$$

Deklaratívne programovanie

Dichotomická diskriminácia

Explicitná definícia funkcie $\max(x, y)$:

$$\begin{aligned}\max(x, y) = {} & \text{case} \\ & \quad x \geq y \Rightarrow x \\ & \quad x \leq y \Rightarrow y \\ & \text{end.}\end{aligned}$$

Program:

$$\begin{aligned}\max(x, y) = {} & \text{case} \\ & \quad (x \geq_* y) = 1 \Rightarrow x \\ & \quad (x \leq_* y) = 1 \Rightarrow y \\ & \text{end.}\end{aligned}$$

Preklad do definície s jednoduchým podmienkovým výrazom:

$$\max(x, y) = \text{if } (x \geq_* y) \neq 0 \text{ then } x \text{ else } y.$$

Podmienkové výrazy

Syntax:

case
$\varphi_1[\vec{x}] \Rightarrow \rho_1[\vec{x}]$
$\vdots$
$\varphi_m[\vec{x}] \Rightarrow \rho_m[\vec{x}]$
end

case
$\chi_1[\vec{x}] = 1 \Rightarrow \rho_1[\vec{x}]$
$\vdots$
$\chi_m[\vec{x}] = 1 \Rightarrow \rho_m[\vec{x}]$
end.

Podmienka úplnosti a jednoznačnosti (výlučnosti):

$$\bigvee_{i=1}^{m} \varphi_i[\vec{x}] \wedge \bigwedge_{i,j=1} \left(\varphi_i[\vec{x}] \wedge \varphi_j[\vec{x}] \rightarrow \rho_i[\vec{x}] = \rho_j[\vec{x}]\right)$$

$$\bigvee_{i=1}^{m} \varphi_i[\vec{x}] \wedge \bigwedge_{\substack{i,j=1\\ i\neq j}}^{m} \neg\left(\varphi_i[\vec{x}] \wedge \varphi_j[\vec{x}]\right).$$

Deklaratívne programovanie

Podmienkové výrazy

Syntax:

$$
\begin{array}{l}
\text{case} \\
\quad \varphi_1[\vec{x}] \Rightarrow \rho_1[\vec{x}] \\
\quad \vdots \\
\quad \varphi_m[\vec{x}] \Rightarrow \rho_m[\vec{x}] \\
\text{end}
\end{array}
\qquad\qquad
\begin{array}{l}
\text{case} \\
\quad \chi_1[\vec{x}] = 1 \Rightarrow \rho_1[\vec{x}] \\
\quad \vdots \\
\quad \chi_m[\vec{x}] = 1 \Rightarrow \rho_m[\vec{x}] \\
\text{end.}
\end{array}
$$

Tu $\chi_i[\vec{x}]$ je charakteristický term pre $\varphi_i[\vec{x}]$:

$$\chi_i[\vec{x}] = 1 \lor \chi_i[\vec{x}] = 0 \qquad \varphi_i[\vec{x}] \leftrightarrow \chi_i[\vec{x}] = 1.$$

Sémantika:

$$\mathrm{D}\Big(\chi_1[\vec{x}], \rho_1[\vec{x}], \ldots, \mathrm{D}\big(\chi_m[\vec{x}], \rho_m[\vec{x}], 0\big) \ldots\Big).$$

Deklaratívne programovanie

Monadická diskriminácia

Rekurzívna definícia umocňovania:

$$
\begin{aligned}
x^y = \mathrm{case}& \\
&y = 0 \Rightarrow 1 \\
&y = z + 1 \Rightarrow x \cdot x^z \\
\mathrm{end}&.
\end{aligned}
$$

Program:

$$
\begin{aligned}
x^y = \mathrm{case}& \\
&(y =_* 0) = 1 \Rightarrow 1 \\
&(y \neq_* 0) = 1 \wedge y \dot{-} 1 = z \Rightarrow x \cdot x^z \\
\mathrm{end}&.
\end{aligned}
$$

Preklad do definície s jednoduchým podmienkovým výrazom:

$$x^y = \mathrm{if}\ (y =_* 0) \neq 0\ \mathrm{then}\ 1\ \mathrm{else}\ x \cdot x^{y \dot{-} 1}.$$

Deklaratívne programovanie

Monadická diskriminácia

Rekurzívna definícia umocňovania:

$$
\begin{aligned}
x^y = {} & \text{case} \\
& \quad y = 0 \Rightarrow 1 \\
& \quad y = z + 1 \Rightarrow x \cdot x^z \\
& \text{end.}
\end{aligned}
$$

Program:

$$
\begin{aligned}
x^y = {} & \text{case} \\
& \quad (y =_* 0) = 1 \Rightarrow 1 \\
& \quad (y \neq_* 0) = 1 \Rightarrow \text{let } z = y \dot{-} 1 \text{ in } x \cdot x^z \\
& \text{end.}
\end{aligned}
$$

Preklad do definície s jednoduchým podmienkovým výrazom:

$$x^y = \text{if } y \neq 0 \text{ then } x \cdot x^{y \dot{-} 1} \text{ else } 1.$$

Párová diskriminácia

Rekurzívna definícia párovej veľkosti prirodzeného čísla:

$$
\begin{aligned}
|x|_{\mathrm{p}} = \mathsf{case}\\
&x = 0 \Rightarrow 0\\
&x = \langle v, w\rangle \Rightarrow |v|_{\mathrm{p}} + |w|_{\mathrm{p}} + 1\\
\mathsf{end}.
\end{aligned}
$$

Program:

$$
\begin{aligned}
|x|_{\mathrm{p}} = \mathsf{case}\\
&(x =_* 0) = 1 \Rightarrow 0\\
&(x \neq_* 0) = 1 \Rightarrow \mathsf{let}\ v = \pi_1(x) \wedge w = \pi_2(x)\ \mathsf{in}\\
&\quad |v|_{\mathrm{p}} + |w|_{\mathrm{p}} + 1\\
\mathsf{end}.
\end{aligned}
$$

Preklad do definície s jednoduchým podmienkovým výrazom:

$$|x|_{\mathrm{p}} = \mathsf{if}\ x \neq 0\ \mathsf{then}\ |\pi_1(x)|_{\mathrm{p}} + |\pi_2(x)|_{\mathrm{p}} + 1\ \mathsf{else}\ 0.$$

Deklaratívne programovanie

Porovnávanie so vzorom ('pattern matching')

Formula $\varphi[\vec{x};\vec{y}]$ spĺňa podmienku jednoznačnosti

$$\varphi[\vec{x};\vec{y}] \land \varphi[\vec{x};\vec{z}] \to \bigwedge_{i=1}^{m} y_i = z_i .$$

Tu $\vec{x}$ resp. $\vec{y}$ sú vstúpne resp. výstupné premenné φ.
Charakteristický term $\chi[\vec{x}]$ pre porovnávanie so vzorom:

$$\chi[\vec{x}] = 1 \lor \chi[\vec{x}] = 0 \qquad \exists \vec{y}\, \varphi[\vec{x};\vec{y}] \leftrightarrow \chi[\vec{x}] = 1.$$

Svedčiace termy $\vec{\omega}[\vec{x}]$ pre výstupné premenné:

$$\exists \vec{y}\, \varphi[\vec{x};\vec{y}] \leftrightarrow \varphi\big[\vec{x};\vec{\omega}[\vec{x}]\big].$$

Platí

$$\varphi[\vec{x};\vec{y}] \leftrightarrow \chi[\vec{x}] = 1 \land \bigwedge_{i=1}^{m} \omega_i[\vec{x}] = y_i .$$

Podmienkové výrazy

Syntax:

case
$\varphi_1[\vec{x};\vec{y}_1] \Rightarrow \rho_1[\vec{x},\vec{y}_1]$
$\vdots$
$\varphi_m[\vec{x};\vec{y}_m] \Rightarrow \rho_m[\vec{x};\vec{y}_m]$
end

case
$\chi_1[\vec{x}] = 1 \wedge \vec{\omega}_1[\vec{x}] = \vec{y}_1 \Rightarrow \rho_1[\vec{x},\vec{y}_1]$
$\vdots$
$\chi_m[\vec{x}] = 1 \wedge \vec{\omega}_m[\vec{x}] = \vec{y}_m \Rightarrow \rho_m[\vec{x},\vec{y}_m]$
end.

Podmienka úplnosti a jednoznačnosti (výlučnosti):

$$\bigvee_{i=1}^{m} \exists \vec{y}_i \varphi_i[\vec{x};\vec{y}_1] \wedge \bigwedge_{i,j=1}^{m} \forall \vec{y}_i \forall \vec{y}_j \big(\varphi_i[\vec{x};\vec{y}_i] \wedge \varphi_j[\vec{x};\vec{y}_j] \to \rho_i[\vec{x},\vec{y}_i] = \rho_j[\vec{x},\vec{y}_j]\big)$$

$$\bigvee_{i=1}^{m} \exists \vec{y}_i \varphi_i[\vec{x};\vec{y}_1] \wedge \bigwedge_{\substack{i,j=1\\ i\neq j}}^{m} \neg\big(\exists \vec{y}_i \varphi_i[\vec{x};\vec{y}_i] \wedge \exists \vec{y}_j \varphi_j[\vec{x};\vec{y}_j]\big).$$

Deklaratívne programovanie

Podmienkové výrazy

Syntax:

$$\begin{array}{ll}
\text{case} & \text{case} \\
\quad \varphi_1[\vec{x}; \vec{y}_1] \Rightarrow \rho_1[\vec{x}, \vec{y}_1] & \quad \chi_1[\vec{x}] = 1 \wedge \vec{\omega}_1[\vec{x}] = \vec{y}_1 \Rightarrow \rho_1[\vec{x}, \vec{y}_1] \\
\quad \vdots & \quad \vdots \\
\quad \varphi_m[\vec{x}; \vec{y}_m] \Rightarrow \rho_m[\vec{x}; \vec{y}_m] & \quad \chi_m[\vec{x}] = 1 \wedge \vec{\omega}_m[\vec{x}] = \vec{y}_m \Rightarrow \rho_m[\vec{x}, \vec{y}_m] \\
\text{end} & \text{end.}
\end{array}$$

$\chi_i[\vec{x}]$ je charakteristický term pre porovnávanie so vzorom $\varphi_i[\vec{x}; \vec{y}_i]$:

$$\chi_i[\vec{x}] = 1 \vee \chi_i[\vec{x}] = 0 \qquad \exists \vec{y}_i \varphi_i[\vec{x}; \vec{y}_i] \leftrightarrow \chi_i[\vec{x}] = 1.$$

Sémantika:

$$\mathrm{D}\Big(\chi_1[\vec{x}], \rho_1[\vec{x}, \vec{\omega}_1[\vec{x}]], \ldots, \mathrm{D}\big(\chi_m[\vec{x}], \rho_m[\vec{x}, \vec{\omega}_m[\vec{x}]], 0\big) \ldots\Big).$$

Podmienkové výrazy

Syntax:

$$
\begin{array}{ll}
\text{case} & \text{case} \\
\quad \varphi_1[\vec{x}; \vec{y}_1] \Rightarrow \rho_1[\vec{x}, \vec{y}_1] & \quad \chi_1[\vec{x}] = 1 \wedge \vec{\omega}_1[\vec{x}] = \vec{y}_1 \Rightarrow \rho_1[\vec{x}, \vec{y}_1] \\
\quad \vdots & \quad \vdots \\
\quad \varphi_m[\vec{x}; \vec{y}_m] \Rightarrow \rho_m[\vec{x}; \vec{y}_m] & \quad \chi_m[\vec{x}] = 1 \wedge \vec{\omega}_m[\vec{x}] = \vec{y}_m \Rightarrow \rho_m[\vec{x}, \vec{y}_m] \\
\text{end} & \text{end.}
\end{array}
$$

Matematická notácia:

$$\mathcal{D}^{\vec{\omega}_1,\ldots,\vec{\omega}_m}_{\chi_1,\ldots,\chi_m}\big(\varphi_1[\vec{x}; \vec{y}_1], \rho_1[\vec{x}, \vec{y}_1], \ldots, \varphi_m[\vec{x}; \vec{y}_m], \rho_m[\vec{x}, \vec{y}_m]\big).$$

Záver

4. cvičenie

- Výsledky s komentárom nájdete v MS Teams.

5. cvičenie

- Začína hneď po prednáške.
- Úlohy odovzdať do MOODLE najneskôr do 18:00 v sobotu tento týždeň.

6. prednáška

- Poza primitívnu rekurziu:
  - Ackermann-Péterovej funkcia,
  - univerzálna funkcia pre primitívne rekurzívne funkcie.

Koniec prednášky