

## Teorija programskih jezikov: 2. izpit

16. februar 2024

### 1. naloga (20 točk)

V  $\lambda$ -računu predstavimo kombinatorje S, K in I z izrazi:

$$s = \lambda x. \lambda y. \lambda z. (xz)(yz)$$

$$k = \lambda x. \lambda y. x$$

$$i = s k k$$

**a) (5 točk)** Naj bo  $V$  poljubna vrednost. Zapišite vse korake v neučakani evalvaciji  $iV \rightsquigarrow^* V$ . Izpeljav posameznih korakov ni treba pisati.

**b) (10 točk)** S pomočjo Hindley-Milnerjevega algoritma izračunajte najbolj splošen tip izraza  $s$ .

**c) (5 točk)** Dokažite, da izraz  $(sii)(sii)$  nima veljavnega tipa.

### 2. naloga (20 točk)

V  $\lambda$ -račun z neučakano semantiko dodamo izraz  $\text{next}$ , ki ob vsakem klicu vrne naslednji člen iz vnaprej danega neskončnega zaporedja celih števil:

$$\text{izraz } e ::= x \mid \underline{n} \mid \text{true} \mid \text{false} \mid e_1 + e_2 \mid e_1 = e_2 \mid \text{if } e \text{ then } e_1 \text{ else } e_2 \mid \lambda x. e \mid e_1 e_2 \mid \text{next}$$

Na primer, izraz  $\text{next} + \text{next} = \text{next}$  izračuna, če je vsota prvih dveh členov enaka tretjemu členu.

Za razširjeni  $\lambda$ -račun zapišite pravila za semantiko malih korakov oblike  $\langle e \mid s \rangle \rightsquigarrow \langle e' \mid s' \rangle$ , da se bo zgornji primer ob zaporedju praštevil 2, 3, 5, 7, ... evalviriral kot

$$\begin{aligned} \langle \text{next} + \text{next} = \text{next} \mid 2, 3, 5, \dots \rangle &\rightsquigarrow \\ \langle 2 + \text{next} = \text{next} \mid 3, 5, 7, \dots \rangle &\rightsquigarrow \\ \langle 2 + 3 = \text{next} \mid 5, 7, 11, \dots \rangle &\rightsquigarrow \\ \langle 5 = \text{next} \mid 5, 7, 11, \dots \rangle &\rightsquigarrow \\ \langle 5 = 5 \mid 7, 11, 13, \dots \rangle &\rightsquigarrow \\ \langle \text{true} \mid 7, 11, 13, \dots \rangle &\end{aligned}$$

### 3. naloga (20 točk)

**a) (5 točk)** Naj bo  $D$  domena in  $f : D \rightarrow D$  zvezna preslikava. Pokažite, da za poljuben  $y \in D$  velja

$$y \geq f(y) \implies y \geq \mu x. f(x)$$

kjer je  $\mu x. f(x)$  alternativen zapis za  $\text{fix}(f)$ , torej najmanjšo fiksno točko preslikave  $f$ .

**b) (15 točk)** Dani naj bosta domeni  $D, E$  ter zvezni preslikavi  $f : D \times E \rightarrow D$  in  $g : D \times E \rightarrow E$ . Definirajmo

$$\begin{aligned} x_0 &= \mu x. f(x, \mu y. g(x, y)) \\ y_0 &= \mu y. g(x_0, y) \end{aligned}$$

Predpostavite lahko, da so vse fiksne točke dobro definirane, torej da naštetih predpisov res podajajo zvezne preslikave.

Pokažite, da sta  $x_0$  in  $y_0$  najmanjši hkratni fiksni točki  $f$  in  $g$ , torej najmanjši vrednosti, da velja  $x_0 = f(x_0, y_0)$  in  $y_0 = g(x_0, y_0)$ .

#### 4. naloga (20 točk)

Programi v jeziku MINIWASM so sestavljeni iz zaporedij ukazov, ki jih podamo s sintakso:

ukaz  $i ::= \text{push}_v \mid \text{drop} \mid \text{add} \mid \text{eq} \mid \text{if } is_1 is_2$   
 program  $is ::= [] \mid i :: is$   
 vrednost  $v ::= \underline{n} \mid \text{true} \mid \text{false}$

MINWASM je skladovni jezik, kar pomeni, da programi ob izvajanju spreminjajo sklad vrednosti  $vs = v_1 :: v_2 :: \dots :: v_k :: []$ . Operacijsko semantiko definiramo z relacijo  $is \mid vs \hookrightarrow is' \mid vs'$ , podano kot

$$\begin{array}{c} \frac{}{(\text{push}_v :: is) \mid vs \hookrightarrow is \mid (v :: vs)} \qquad \frac{}{(\text{drop} :: is) \mid (v :: vs) \hookrightarrow is \mid vs} \\[10pt] \frac{}{(\text{add} :: is) \mid (\underline{n_1} :: \underline{n_2} :: vs) \hookrightarrow is \mid (\underline{n_1 + n_2} :: vs)} \qquad \frac{n_1 = n_2}{(\text{eq} :: is) \mid (\underline{n_1} :: \underline{n_2} :: vs) \hookrightarrow is \mid (\text{true} :: vs)} \\[10pt] \frac{n_1 \neq n_2}{(\text{eq} :: is) \mid (\underline{n_1} :: \underline{n_2} :: vs) \hookrightarrow is \mid (\text{false} :: vs)} \qquad \frac{}{((\text{if } is_1 is_2) :: is) \mid (\text{true} :: vs) \hookrightarrow (is_1 @ is) \mid vs} \\[10pt] \frac{}{((\text{if } is_1 is_2) :: is) \mid (\text{false} :: vs) \hookrightarrow (is_2 @ is) \mid vs} \end{array}$$

kjer je @ stikanje seznamov.

Če vrednostim in skladom na očitni način z relacijama  $\vdash v : t$  in  $\vdash vs : ts$  priredimo tipe

tip vrednosti  $t ::= \text{int} \mid \text{bool}$   
 tip sklada  $ts ::= [] \mid t :: ts$

lahko definiramo tromestni relaciji  $\vdash i : ts_1 \rightarrow ts_2$  in  $\vdash is : ts_1 \rightarrow ts_2$ , ki povesta, da ukaz  $i$  oziroma program  $is$  sklad tipa  $ts_1$  spremenita v sklad tipa  $ts_2$ :

$$\begin{array}{c} \frac{\vdash v : t}{\vdash \text{push}_v : ts \rightarrow (t :: ts)} \qquad \frac{}{\vdash \text{drop} : (t :: ts) \rightarrow ts} \qquad \frac{}{\vdash \text{add} : (\text{int} :: \text{int} :: ts) \rightarrow (\text{int} :: ts)} \\[10pt] \frac{}{\vdash \text{eq} : (\text{int} :: \text{int} :: ts) \rightarrow (\text{bool} :: ts)} \qquad \frac{\vdash is_1 : ts_1 \rightarrow ts_2 \quad \vdash is_2 : ts_1 \rightarrow ts_2}{\vdash (\text{if } is_1 is_2) : (\text{bool} :: ts_1) \rightarrow ts_2} \\[10pt] \frac{}{\vdash [] : ts \rightarrow ts} \qquad \frac{\vdash i : ts_1 \rightarrow ts_2 \quad \vdash is : ts_2 \rightarrow ts_3}{\vdash (i :: is) : ts_1 \rightarrow ts_3} \end{array}$$

Zapišite trditvi o ohranitvi in napredku ter eno od njiju dokažite.