Programmikeelte semantika 2012
Koduülesanded 1

Mart Karu

Aprill 2012

1 Ülesanne 1

1.1 Loomulikus operatsioonisemantikas

$$\frac{\dfrac{\langle y := y - x, s_0\rangle \to s_1}{\langle \text{if } x \le y \text{ then } y := y - x \text{ else } x := x - y, s_0\rangle \to s_1}\ T_1}{\langle \text{while } x \neq y \text{ do (if } x \le y \text{ then } y := y - x \text{ else } x := x - y), s_0\rangle \to s_4}$$

$$T_1 = \frac{\dfrac{\langle x := x - y, s_1\rangle \to s_2}{\langle \text{if } x \le y \text{ then } y := y - x \text{ else } x := x - y, s_1\rangle \to s_2}\ T_2}{\langle \text{while } x \neq y \text{ do (if } x \le y \text{ then } y := y - x \text{ else } x := x - y), s_1\rangle \to s_4}$$

$$T_2 = \frac{\dfrac{\langle x := x - y, s_2\rangle \to s_3}{\langle \text{if } x \le y \text{ then } y := y - x \text{ else } x := x - y, s_2\rangle \to s_3}\ T_3}{\langle \text{while } x \neq y \text{ do (if } x \le y \text{ then } y := y - x \text{ else } x := x - y), s_2\rangle \to s_4}$$

$$T_3 = \frac{\dfrac{\langle y := y - x, s_3\rangle \to s_4}{\langle \text{if } x \le y \text{ then } y := y - x \text{ else } x := x - y, s_3\rangle \to s_4}}{\langle \text{while } x \neq y \text{ do (if } x \le y \text{ then } y := y - x \text{ else } x := x - y), s_1\rangle \to s_4}$$

$$s_0 = [x \mapsto 30, y \mapsto 42]$$

$$s_1 = [x \mapsto 30, y \mapsto 12]$$

$$s_2 = [x \mapsto 18, y \mapsto 12]$$

$$s_3 = [x \mapsto 6, y \mapsto 12]$$

$$s_4 = [x \mapsto 6, y \mapsto 6]$$

1.2 struktuurses operatsioonisemantikas

$$\langle \text{while } x \neq y \text{ do (if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s\rangle$$

$$(1) \Rightarrow \langle \text{while } x \neq y \text{ do (if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s[y \mapsto 12]\rangle$$

$$(2) \Rightarrow \langle \text{while } x \neq y \text{ do (if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s[y \mapsto 12][x \mapsto 18]\rangle$$

$$(3) \Rightarrow \langle \text{while } x \neq y \text{ do (if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s[y \mapsto 12][x \mapsto 6]\rangle$$

$$(4) \Rightarrow \langle \text{while } x \neq y \text{ do (if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s[y \mapsto 6][x \mapsto 6]\rangle$$

$$(5) \Rightarrow s[y \mapsto 6][x \mapsto 6]$$

$(1) : \langle \text{while } x \neq y \text{ do (if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s\rangle \Rightarrow$

$\langle \text{if } x \neq y \text{ then ((if } x \leq y \text{ then } y := y - x \text{ else } x := x - y)); \text{ while } x \neq y \text{ do}$
$(\text{if } x \leq y \text{ then } y := y - x \text{ else } x := x - y)) \text{ else skip}, s\rangle \Rightarrow$

$$\frac{\langle \text{if } x \leq y \text{ then } y := y - x \text{ else } x := x - y, s\rangle \Rightarrow \langle y := y - x, s\rangle}{\langle \text{if } x \leq y \text{ then } y := y - x \text{ else } x := x - y; \text{ while } x \neq y \text{ do}}$$
$(\text{if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s\rangle \Rightarrow$
$\langle y := y - x; \text{ while } x \neq y \text{ do (if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s\rangle \Rightarrow$ *program@epstopdf*

$$\frac{\langle y := y - x, s\rangle \Rightarrow s[y \mapsto 12]}{\langle y := y - x; \text{ while } x \neq y \text{ do (if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s\rangle \Rightarrow}$$
$\langle \text{while } x \neq y \text{ do if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s[y \mapsto 12]\rangle$ *program@epstopdf*

$(2) : \langle \text{while } x \neq y \text{ do (if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s[y \mapsto 12]\rangle \Rightarrow$

$\langle \text{if } x \neq y \text{ then ((if } x \leq y \text{ then } y := y - x \text{ else } x := x - y)); \text{ while } x \neq y \text{ do}$
$(\text{if } x \leq y \text{ then } y := y - x \text{ else } x := x - y)) \text{ else skip}, s[y \mapsto 12]\rangle \Rightarrow$

$$\frac{\langle \text{if } x \leq y \text{ then } y := y - x \text{ else } x := x - y, s[y \mapsto 12]\rangle \Rightarrow \langle x := x - y, s[y \mapsto 12]\rangle}{\langle \text{if } x \leq y \text{ then } y := y - x \text{ else } x := x - y; \text{ while } x \neq y \text{ do}}$$
$(\text{if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s[y \mapsto 12]\rangle \Rightarrow$
$\langle x := x - y; \text{ while } x \neq y \text{ do (if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s[y \mapsto 12]\rangle \Rightarrow$ *program@epstopd*

$$\frac{\langle x := x - y, s[y \mapsto 12]\rangle \Rightarrow s[y \mapsto 12][x \mapsto 18]}{\langle x := x - y; \text{ while } x \neq y \text{ do (if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s[y \mapsto 12]\rangle \Rightarrow}$$
$\langle \text{while } x \neq y \text{ do if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s[y \mapsto 12][x \mapsto 18]\rangle$ *program@epstopd*

(3) :$\langle$while $x \neq y$ do (if $x \leq y$ then $y := y - x$ else $x := x - y$), $s[y \mapsto 12][x \mapsto 18]\rangle \Rightarrow$

$\langle$if $x \neq y$ then ((if $x \leq y$ then $y := y - x$ else $x := x - y$)); while $x \neq y$ do
(if $x \leq y$ then $y := y - x$ else $x := x - y$)) else skip, $s[y \mapsto 12][x \mapsto 18]\rangle \Rightarrow$

$$\frac{\langle \text{if } x \leq y \text{ then } y := y - x \text{ else } x := x - y, s[y \mapsto 12][x \mapsto 18]\rangle \Rightarrow \langle x := x - y, s[y \mapsto 12][x \mapsto 18]\rangle}{\langle \text{if } x \leq y \text{ then } y := y - x \text{ else } x := x - y; \text{ while } x \neq y \text{ do}}$$

(if $x \leq y$ then $y := y - x$ else $x := x - y$), $s[y \mapsto 12][x \mapsto 18]\rangle \Rightarrow$
$\langle x := x - y;$ while $x \neq y$ do (if $x \leq y$ then $y := y - x$ else $x := x - y$), $s[y \mapsto 12][x \mapsto 18]\rangle \Rightarrow$ *progra*

$$\frac{\langle x := x - y, s[y \mapsto 12][x \mapsto 18]\rangle \Rightarrow s[y \mapsto 12][x \mapsto 6]}{\langle x := x - y; \text{ while } x \neq y \text{ do (if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s[y \mapsto 12][x \mapsto 18]\rangle \Rightarrow}$$

$\langle$while $x \neq y$ do if $x \leq y$ then $y := y - x$ else $x := x - y$), $s[y \mapsto 12][x \mapsto 6]\rangle$ *progra*

(4) :$\langle$while $x \neq y$ do (if $x \leq y$ then $y := y - x$ else $x := x - y$), $s[y \mapsto 12][x \mapsto 6]\rangle \Rightarrow$

$\langle$if $x \neq y$ then ((if $x \leq y$ then $y := y - x$ else $x := x - y$)); while $x \neq y$ do
(if $x \leq y$ then $y := y - x$ else $x := x - y$)) else skip, $s[y \mapsto 12][x \mapsto 6]\rangle \Rightarrow$

$$\frac{\langle \text{if } x \leq y \text{ then } y := y - x \text{ else } x := x - y, s[y \mapsto 12][x \mapsto 6]\rangle \Rightarrow \langle y := y - x, s[y \mapsto 12][x \mapsto 6]\rangle}{\langle \text{if } x \leq y \text{ then } y := y - x \text{ else } x := x - y; \text{ while } x \neq y \text{ do}}$$

(if $x \leq y$ then $y := y - x$ else $x := x - y$), $s[y \mapsto 12][x \mapsto 6]\rangle \Rightarrow$
$\langle y := y - x;$ while $x \neq y$ do (if $x \leq y$ then $y := y - x$ else $x := x - y$), $s[y \mapsto 12][x \mapsto 6]\rangle \Rightarrow$ *program*

$$\frac{\langle y := y - x, s[y \mapsto 12][x \mapsto 6]\rangle \Rightarrow s[y \mapsto 6][x \mapsto 6]}{\langle y := y - x; \text{ while } x \neq y \text{ do (if } x \leq y \text{ then } y := y - x \text{ else } x := x - y), s[y \mapsto 12][x \mapsto 18]\rangle \Rightarrow}$$

$\langle$while $x \neq y$ do if $x \leq y$ then $y := y - x$ else $x := x - y$), $s[y \mapsto 6][x \mapsto 6]\rangle$ *program*

(5) :$\langle$while $x \neq y$ do (if $x \leq y$ then $y := y - x$ else $x := x - y$), $s[y \mapsto 6][x \mapsto 6]\rangle \Rightarrow$

$\langle$if $x \neq y$ then ((if $x \leq y$ then $y := y - x$ else $x := x - y$)); while $x \neq y$ do
(if $x \leq y$ then $y := y - x$ else $x := x - y$)) else skip, $s[y \mapsto 6][x \mapsto 6]\rangle \Rightarrow$

$\langle$skip, $s[y \mapsto 6][x \mapsto 6]\rangle \Rightarrow s[y \mapsto 6][x \mapsto 6]$

2 Ülesanne 2

2.1 a

Kui $b_1 =$ false ja $b_2 =$ false ning $b_1 = false$ ja $b_2 =$ true:

$$\frac{\langle S_2, s\rangle \to s'}{\langle \text{if } (b_1 \wedge b_2) \text{ then } S_1 \text{ else } S_2, s\rangle \to s'}$$

Millest annab tuletada:

$$\frac{\langle S_2, s\rangle \to s'}{\langle \text{if } b_1 \text{ then } (\text{if } b_2 \text{ then } S_1 \text{ else } S_2)) \text{ else } S_2, s\rangle \to s'}$$

Teistpidi:

$$\frac{\langle S_2, s\rangle \to s'}{\langle \text{if } b_1 \text{ then } (\text{if } b_2 \text{ then } S_1 \text{ else } S_2)) \text{ else } S_2, s\rangle \to s'}$$

Millest annab samuti tuletada:

$$\frac{\langle S_2, s\rangle \to s'}{\langle \text{if } (b_1 \wedge b_2) \text{ then } S_1 \text{ else } S_2, s\rangle \to s'}$$

Kui $b_1 =$ true ja $b_2 =$ false:

$$\frac{\langle S_2, s\rangle \to s'}{\langle \text{if } (b_1 \wedge b_2) \text{ then } S_1 \text{ else } S_2, s\rangle \to s'}$$

Millest annab tuletada:

$$\frac{\dfrac{\langle S_2, s\rangle \to s'}{\langle \text{if } b_2 \text{ then } S_1 \text{ else } S_2), s\rangle \to s'}}{\langle \text{if } b_1 \text{ then } (\text{if } b_2 \text{ then } S_1 \text{ else } S_2)) \text{ else } S_2, s\rangle \to s'}$$

Teistpidi:

$$\frac{\dfrac{\langle S_2, s\rangle \to s'}{\langle \text{if } b_2 \text{ then } S_1 \text{ else } S_2), s\rangle \to s'}}{\langle \text{if } b_1 \text{ then } (\text{if } b_2 \text{ then } S_1 \text{ else } S_2)) \text{ else } S_2, s\rangle \to s'}$$

Millest annab samuti tuletada:

$$\frac{\langle S_2, s\rangle \to s'}{\langle \text{if } (b_1 \wedge b_2) \text{ then } S_1 \text{ else } S_2, s\rangle \to s'}$$

Kui $b_1 =$ true ja $b_2 =$ true:

$$\frac{\langle S_1, s\rangle \to s'}{\langle \text{if } (b_1 \wedge b_2) \text{ then } S_1 \text{ else } S_2, s\rangle \to s'}$$

Millest annab tuletada:

$$\frac{\dfrac{\langle S_1, s\rangle \to s'}{\langle \text{if } b_2 \text{ then } S_1 \text{ else } S_2), s\rangle \to s'}}{\langle \text{if } b_1 \text{ then } (\text{if } b_2 \text{ then } S_1 \text{ else } S_2)) \text{ else } S_2, s\rangle \to s'}$$

Teistpidi:

$$\frac{\frac{\langle S_1, s\rangle \to s'}{\langle \text{if } b_2 \text{ then } S_1 \text{ else } S_2), s\rangle \to s'}}{\langle \text{if } b_1 \text{ then (if } b_2 \text{ then } S_1 \text{ else } S_2)) \text{ else } S_2, s\rangle \to s'}$$

Millest annab samuti tuletada:

$$\frac{\langle S_1, s\rangle \to s'}{\langle \text{if } (b_1 \wedge b_2) \text{ then } S_1 \text{ else } S_2, s\rangle \to s'}$$

2.2 b

2.2.1 Kui $b_1 = false$ ja $b_2 = false$ ning $b_1 = false$ ja $b_2 = true$, siis

$$\langle \text{while } b_1 \text{ do (while } (b_1 \wedge b_2) \text{ do } S), s\rangle \to s$$

While-tsükli keha ei täideta ning olek s ei muutu. Sama juhtub ka teise avaldisega:

$$\langle \text{while } b_1 \text{ do (if } b_2 \text{ then } S \text{ else (while true do skip))}, s\rangle \to s$$

2.2.2 Kui $b_1 = true$ ja $b_2 = false$

$$\frac{T_1 \quad T_2}{\langle \text{while } b_1 \text{ do (while } (b_1 \wedge b_2) \text{ do } S), s\rangle \to s}$$

$$T_1 = \langle \text{while } (b_1 \wedge b_2) \text{ do } S, s\rangle \to s$$

$$T_2 = \langle \text{while } b_1 \text{ do (while } (b_1 \wedge b_2)), s\rangle \to s$$

Kuna T_1 tsükli keha ei läbita, jääb olek s muutmata ning T_1 saab võrdsustada oma lõpptulemuse osas käsuga skip. Samuti võib sisuliselt T_2 asendada lausega:

$$T_2 = \langle \text{while } b_1 \text{ do skip}), s\rangle \to s$$

Kuna $b_1 = true$, siis saab sellest ehitada teise avaldise tuletuspuu:

$$\frac{\frac{\langle \text{while true do skip}, s\rangle}{\langle \text{if } b2 \text{ then } S \text{ else (while true do skip))}, s\rangle} \; T_3}{\langle \text{while } b1 \text{ do (if } b2 \text{ then } S \text{ else (while true do skip))}, s\rangle \to s}$$

$$T_3 = \langle \text{while } b1 \text{ do (if } b2 \text{ then } S \text{ else (while true do skip))}, s\rangle \to s$$

2.2.3 Kui $b_1 = true$ ja $b_2 = true$

Esimene avaldis:

$$\frac{\frac{\frac{T_1}{\langle S, s\rangle \to s_1} \quad \frac{T_2}{\langle \text{while } (b_1 \wedge b_2) \text{ do } S, s_1\rangle \to s_2}}{\langle \text{while } (b_1 \wedge b_2) \text{ do } S, s\rangle \to s_2} \quad \langle \text{while } b_1 \text{ do (while } (b_1 \wedge b_2) \text{ do } S), s_2\rangle \to s'}{\langle \text{while } b_1 \text{ do (while } (b_1 \wedge b_2) \text{ do } S), s\rangle \to s'}$$

Kasutades tuletuspuid T_1 ja T_2 ja eeldades, et $(b_1 \wedge b_2)$ asemel saab kasutada while tingimuseks ainult b_1 saab ehitada teise avaldise:

$$\dfrac{\dfrac{\dfrac{T_1}{\langle S, s\rangle \to s_1}}{\langle \text{if } b2 \text{ then } S \text{ else (while true do skip)}, s\rangle \to s_1} \quad \dfrac{\dfrac{T_2}{\langle \text{while } b_1 \text{ do } S, s_1\rangle \to s'}}{\langle \text{while } b1 \text{ do (if } b2 \text{ then } S \text{ else (while true do skip))}, s_1\rangle \to s'}}{\langle \text{while } b1 \text{ do (if } b2 \text{ then } S \text{ else (while true do skip))}, s\rangle \to s'}$$

2.3 c

Esimene avaldis:

$$\dfrac{\dfrac{s[x \mapsto A[\![a_1]\!]s] = s'}{\langle x := a_1, s\rangle \to s'} \quad \dfrac{s'[y \mapsto A[\![a_2]\!]s'] = s''}{\langle y := a_2, s'\rangle \to s''}}{\langle x := a_1; y := a_2, s\rangle \to s''}$$

Esimese avaldise tuletuspuust saab koostada teise avaldise. Oleku uuendamisel tehtud omistuste järjekorda võib muuta, sest x ei leia kasutamist avaldises a_2 ning y ei leia kasutamist avaldises a_1

$$\dfrac{\dfrac{s[y \mapsto A[\![a_2]\!]s] = s'}{\langle y := a_2, s\rangle \to s'} \quad \dfrac{s'[x \mapsto A[\![a_1]\!]s'] = s''}{\langle x := a_1, s'\rangle \to s''}}{\langle y := a_2; x := a_1, s\rangle \to s''}$$

3 Ülesanne 3

3.1 Osa 1 - $b?a_0 : a_1$

Aritmeetikaavaldise $b?a_0 : a_1$ toetamiseks on tarvis lisada järgmised reeglid:

$$\dfrac{\langle a_0, s\rangle \to_{\text{Aexp}} z}{\langle b?a_0 : a_1, s\rangle \to_{\text{Aexp}} z} \text{if} B[\![b]\!] = \text{tt}$$

ja

$$\dfrac{\langle a_1, s\rangle \to_{\text{Aexp}} z}{\langle b?a_0 : a_1, s\rangle \to_{\text{Aexp}} z} \text{ if } B[\![b]\!] = \text{ff}$$

3.2 Osa 2 - $S \triangleright a$

Aritmeetikaavaldise $S \triangleright a$ toetamiseks tuleb kõigepealt käivitada statemnt S etteantud olekuga s, mille tulemusena tekib olek s'. Olekut s' kasutatakse aritmeetikaavaldise a rehkendamises, mis väljastab arvu z. Olek ei kandu edasi, mistõttu on S käskude mõju globaalsele olekule mittemõjuv.

$$\dfrac{\langle S, s\rangle \to s', \quad \langle a, s'\rangle \to_{\text{Aexp}} z}{\langle S \triangleright a, s\rangle \to_{\text{Aexp}} z}$$

4 Ülesanne 4

Nelikul $\langle a, s\rangle \to_{\text{Aexp}} \langle z, s'\rangle$ põhineva keele loomuliku semantika reeglid on:

4.1 Aritmeetikaavaldised

$$\langle n, s\rangle \to_{\text{Aexp}} \langle N[\![n]\!], s\rangle$$

$$\langle x, s\rangle \to_{\text{Aexp}} \langle s\ x, s\rangle$$

$$\langle x := a, s\rangle \to_{\text{Aexp}} \langle z, s'[x \mapsto z]\rangle$$

z on avaldise a arvutamisel saadud tulemus ning s' on avaldise a arvutamisel tekkinud olek.

$$\frac{\langle a_0, s\rangle \to_{\text{Aexp}} \langle z_0, s'\rangle, \langle a_1, s'\rangle \to_{\text{Aexp}} \langle z_1, s''\rangle}{\langle a_0 + a_1, s\rangle \to_{\text{Aexp}} \langle z, s''\rangle}, \text{ kus } z = z_0 + z_1$$

$$\frac{\langle a_0, s\rangle \to_{\text{Aexp}} \langle z_0, s'\rangle, \langle a_1, s'\rangle \to_{\text{Aexp}} \langle z_1, s''\rangle}{\langle a_0 - a_1, s\rangle \to_{\text{Aexp}} \langle z, s''\rangle}, \text{ kus } z = z_0 - z_1$$

$$\frac{\langle a_0, s\rangle \to_{\text{Aexp}} \langle z_0, s'\rangle, \langle a_1, s'\rangle \to_{\text{Aexp}} \langle z_1, s''\rangle}{\langle a_0 * a_1, s\rangle \to_{\text{Aexp}} \langle z, s''\rangle}, \text{ kus } z = z_0 * z_1$$

4.2 Loogikaavaldised

$$\langle true, s\rangle \to_{\text{Bexp}} \langle tt, s\rangle$$

$$\langle false, s\rangle \to_{\text{Bexp}} \langle ff, s\rangle$$

$$\frac{\langle a_0, s\rangle \to_{\text{Aexp}} \langle z_0, s'\rangle, \langle a_1, s'\rangle \to_{\text{Aexp}} \langle z_1, s''\rangle}{\langle a_0 = a_1, s\rangle \to_{\text{Bexp}} \langle ff, s''\rangle}, \text{ if } z_0 \neq z_1$$

$$\frac{\langle a_0, s\rangle \to_{\text{Aexp}} \langle z_0, s'\rangle, \langle a_1, s'\rangle \to_{\text{Aexp}} \langle z_1, s''\rangle}{\langle a_0 = a_1, s\rangle \to_{\text{Bexp}} \langle tt, s''\rangle}, \text{ if } z_0 = z_1$$

$$\frac{\langle a_0, s\rangle \to_{\text{Aexp}} \langle z_0, s'\rangle, \langle a_1, s'\rangle \to_{\text{Aexp}} \langle z_1, s''\rangle}{\langle a_0 \leq a_1, s\rangle \to_{\text{Bexp}} \langle tt, s''\rangle}, \text{ if } z_0 \leq z_1$$

$$\frac{\langle a_0, s\rangle \to_{\text{Aexp}} \langle z_0, s'\rangle, \langle a_1, s'\rangle \to_{\text{Aexp}} \langle z_1, s''\rangle}{\langle a_0 > a_1, s\rangle \to_{\text{Bexp}} \langle ff, s''\rangle} \text{ if } z_0 > z_1$$

$$\langle \neg b, s\rangle \to_{\text{Bexp}} \langle tt, s'\rangle \quad \text{if } bv = ff$$

$$\langle \neg b, s\rangle \to_{\text{Bexp}} \langle ff, s'\rangle \quad \text{if } bv = tt$$

bv on loogikaavaldise b arvutamise tulemus ja s' loogikaavalise arvutamisel muutunud olek

$$\frac{\langle b_0, s\rangle \to_{\text{Bexp}} \langle bv_0, s'\rangle, \langle b_1, s'\rangle \to_{\text{Bexp}} \langle bv_1, s''\rangle}{\langle b_0 \wedge b_1, s\rangle \to_{\text{Bexp}} \langle bv, s''\rangle}, \text{ kus } bv = bv_0 \text{ and } bv_1$$

bv_0 ja bv_1 on loogikaavaldiste b_0 ja b_1 arvutamise tulemused.

4.3 Käsud

$$\frac{\langle a, s\rangle \to_{\text{Aexp}} \langle z, s'\rangle}{\langle x := a, s\rangle \to s'[x \mapsto z]}, \text{ kus } z \text{ on avaldise } a \text{ arvutamise tulemus}$$

$$\frac{\langle b, s\rangle \to_{\text{Bexp}} \langle bv, s'\rangle \quad \langle S_1, s'\rangle \to s''}{\langle \text{if } b \text{ then } S_1 \text{ else } S_2, s\rangle \to s''}, \text{ if } bv = tt \text{ ja } bv \text{ arvutatakse avaldisest } b$$

$$\frac{\langle b, s\rangle \to_{\text{Bexp}} \langle bv, s'\rangle \quad \langle S_2, s'\rangle \to s''}{\langle \text{if } b \text{ then } S_1 \text{ else } S_2, s\rangle \to s''}, \text{ if } bv = ff \text{ ja } bv \text{ arvutatakse avaldisest } b$$

$$\frac{\langle b, s\rangle \to_{\text{Bexp}} \langle bv, s'\rangle \quad \langle S, s'\rangle \to s'' \quad \langle \text{while } b \text{ do } S, s''\rangle \to s'''}{\langle \text{while } b \text{ do } S, s\rangle \to s'''}, \text{ if } bv = tt \text{ ja } bv \text{ arvutatakse avaldisest } b$$

$$\frac{\langle b, s\rangle \to_{\text{Bexp}} \langle bv, s'\rangle}{\langle \text{while } b \text{ do } S, s\rangle \to s'}, \text{ if } bv = ff \text{ ja } bv \text{ arvutatakse avaldisest } b$$

5 Ülesanne 5

5.1 *reset* - käsuta variatsioon loendamisest

$$\langle x := a, s\rangle \to \langle 1, s[x \mapsto A[\![a]\!]s]\rangle$$

$$\langle skip, s\rangle \to \langle 1, s\rangle$$

$$\frac{\langle S_1, s\rangle \to \langle n, s'\rangle \quad \langle S_2, s'\rangle \to \langle n', s''\rangle}{\langle S_1; S_2, s\rangle \to \langle n + n', s''\rangle}$$

$$\frac{\langle S_1, s\rangle \to \langle n, s'\rangle}{\langle \text{if } b \text{ then } S_1 \text{ else } S_2, s\rangle \to \langle n + 1, s'\rangle}, \text{ if } B[\![b]\!]s = tt$$

$$\frac{\langle S_2, s\rangle \to \langle n, s'\rangle}{\langle \text{if } b \text{ then } S_1 \text{ else } S_2, s\rangle \to \langle n + 1, s'\rangle}, \text{ if } B[\![b]\!]s = ff$$

$$\frac{\langle S, s\rangle \to \langle n, s'\rangle \quad \langle \text{while } b \text{ do } S, s'\rangle \to \langle n', s''\rangle}{\langle \text{while } b \text{ do } S, s\rangle \to \langle n + n' + 1, s''\rangle}, \text{ if } B[\![b]\!]s = tt$$

$$\langle \text{while } b \text{ do } S, s\rangle \to \langle 1, s\rangle, \text{ if } B[\![b]\!]s = ff$$

5.2 *reset* - käsuga loendamine

$$\langle x := a, n, s\rangle \to \langle n + 1, s[x \mapsto A[\![a]\!]s]\rangle$$

$$\langle skip, n, s\rangle \to \langle n + 1, s\rangle$$

$$\langle reset, n, s\rangle \to \langle 0, s\rangle$$

$$\frac{\langle S_1, n, s\rangle \to \langle n', s'\rangle \quad \langle S_2, n', s'\rangle \to \langle n'', s''\rangle}{\langle S_1; S_2, s\rangle \to \langle n'', s''\rangle}$$

$$\frac{\langle S_1, n, s\rangle \to \langle n', s'\rangle}{\langle \text{if } b \text{ then } S_1 \text{ else } S_2, n, s\rangle \to \langle n'+1, s'\rangle}, \text{ if } B[\![b]\!]s = tt$$

$$\frac{\langle S_2, n, s\rangle \to \langle n', s'\rangle}{\langle \text{if } b \text{ then } S_1 \text{ else } S_2, n, s\rangle \to \langle n'+1, s'\rangle}, \text{ if } B[\![b]\!]s = ff$$

$$\frac{\langle S, n, s\rangle \to \langle n', s'\rangle \quad \langle \text{while } b \text{ do } S, n', s'\rangle \to \langle n'', s''\rangle}{\langle \text{while } b \text{ do } S, n, s\rangle \to \langle n''+1, s''\rangle}, \text{ if } B[\![b]\!]s = tt$$

$$\langle \text{while } b \text{ do } S, n, s\rangle \to \langle n+1, s\rangle, \text{ if } B[\![b]\!]s = ff$$

6 Ülesanne 6

1. Mõlemad käsud ei lõpeta (tehakse kaks sammu esimesest ja üks teisest, ning tulemuseks on mõlema puhul järgnevad sammud):

$$\frac{\langle S_1, s\rangle \Rightarrow \langle S_1', s'\rangle, \langle S_1', s'\rangle \Rightarrow \langle S_1'', s''\rangle, \langle S_2, s''\rangle \Rightarrow \langle S_2', s'''\rangle}{\langle S_1 \text{ par } S_2, s\rangle \Rightarrow \langle S_1'' \text{ par } S_2', s'''\rangle}$$

2. Esimene käsk jõuab lõppu teise sammu järel, jätkakse teise käsuga:

$$\frac{\langle S_1, s\rangle \Rightarrow \langle S_1', s'\rangle, \langle S_1', s'\rangle \Rightarrow s'', \langle S_2, s''\rangle \Rightarrow \langle S_2', s'''\rangle}{\langle S_1 \text{ par } S_2, s\rangle \Rightarrow \langle S_2', s'''\rangle}$$

3. Esimene käsk jõuab lõppu esimese sammu järel, jätkakse teise käsuga:

$$\frac{\langle S_1, s\rangle \Rightarrow s', \langle S_2, s'\rangle \Rightarrow \langle S_2', s''\rangle}{\langle S_1 \text{ par } S_2, s\rangle \Rightarrow \langle S_2', s''\rangle}$$

4. Teine käsk lõpetab, esimene kestab edasi:

$$\frac{\langle S_1, s\rangle \Rightarrow \langle S_1', s'\rangle, \langle S_1', s'\rangle \Rightarrow \langle S_1'', s''\rangle, \langle S_2, s''\rangle \Rightarrow s'''}{\langle S_1 \text{ par } S_2, s\rangle \Rightarrow \langle S_1'', s'''\rangle}$$

5. Esimene käsk jõuab lõppu teise sammu järel, lõppu jõuab ka teine käsk:

$$\frac{\langle S_1, s\rangle \Rightarrow \langle S_1', s'\rangle, \langle S_1', s'\rangle \Rightarrow s'', \langle S_2, s''\rangle \Rightarrow s'''}{\langle S_1 \text{ par } S_2, s\rangle \Rightarrow s'''}$$

5. Esimene käsk jõuab lõppu teise sammu järel, lõppu jõuab ka teine käsk:

$$\frac{\langle S_1, s\rangle \Rightarrow \langle S_1', s'\rangle, \langle S_2, s'\rangle \Rightarrow s''}{\langle S_1 \text{ par } S_2, s\rangle \Rightarrow s''}$$

7 Ülesanne 7

Abstraktse masina semantika täiendused on:
$\langle$DUP:$c,\ z$:$e,\ s\rangle \triangleright \langle c,\ z$:$z$:$e,\ s\rangle$, kus $z \in (Z \cup T)$
$\langle$SWAP:$c,\ z_1$:z_2:$e,\ s\rangle \triangleright \langle c,\ z_2$:$z_1$:$e,\ s\rangle$, kus $z \in (Z \cup T)$

Osa 2: Näita, et
$CS[\![x := z - y]\!] : CS[\![w := x * z]\!]$ ja
`FETCH z : DUP : FETCH y : SWAP : SUB : DUP : STORE x : MUL : STORE w`
on samaväärsed.
Esimesest koodilõigust saab kompileerida:

$CS[\![x := z - y]\!] : CS[\![w := x * z]\!] \Rightarrow$
$CA[\![z - y]\!]$: `STORE x` : $CA[\![x * z]\!]$: `STORE w` $\Rightarrow$
$CA[\![y]\!] : CA[\![z]\!]$: `SUB` : `STORE x` : $CA[\![z]\!] : CA[\![x]\!]$: `MUL` : `STORE w` $\Rightarrow$
`FETCH y` : `FETCH z` : `SUB` : `STORE x` : `FETCH z` : `FETCH x` : `MUL` : `STORE w`

Samaväärsuse tõestus:
Esimesest koodilõiku interpreteerides saab selgitada selle mõju olekule:

`FETCH y`	$e = [y]$	$s = []$
`FETCH z`	$e = [z, y]$	$s = []$
`SUB`	$e = [(z - y)]$	$s = []$
`STORE x`	$e = []$	$s = [x \mapsto (z - y)]$
`FETCH z`	$e = [z]$	$s = [x \mapsto (z - y)]$
`FETCH x`	$e = [x, z]$	$s = [x \mapsto (z - y)]$
`MUL`	$e = [(x * z)]$	$s = [x \mapsto (z - y)]$
`STORE w`	$e = []$	$s = [x \mapsto (z - y), w \mapsto (x * z)] = [x \mapsto (z - y), w \mapsto ((z - y) * z)]$

`FETCH z`	$e = [z]$	$s = []$
`DUP`	$e = [z, z]$	$s = []$
`FETCH y`	$e = [y, z, z]$	$s = []$
`SWAP`	$e = [z, y, z]$	$s = []$
`SUB`	$e = [(z - y), z]$	$s = []$
`DUP`	$e = [(z - y), (z - y), z]$	$s = []$
`STORE x`	$e = [(z - y), z]$	$s = [x \mapsto (z - y)]$
`MUL`	$e = [((z - y) * z)]$	$s = [x \mapsto (z - y)]$
`STORE w`	$e = []$	$s = [x \mapsto (z - y), w \mapsto ((z - y) * z)]$

Mõlema koodilõigu käivitamisel saadud lõpptulemused (e ja s) on võrdsed, mistõttu võib väita, et koodilõigud on samatähenduslikud käivitamise lõpptulemuse osas.