Семантика на езиците за програмиране - записки

Стефан Вътев[1]

17 януари 2024 г.

[1] stefanv@fmi.uni-sofia.bg

Съдържание

1 Теоретични основи **3**
1.1 Частични наредби 4
1.2 Конструкции на области на Скот 7
1.2.1 Плоска област на Скот 7
1.2.2 Крайно произведение 8
1.3 Изображения в области на Скот 10
1.3.1 Монотонни изображения 11
1.3.2 Непрекъснати изображения 13
1.4 Област на Скот от непрекъснати изображения 16
1.5 Основни непрекъснати изображения 21
1.6 Най-малки неподвижни точки 27
1.7 Оператор за най-малка неподвижна точка 33
1.8 Най-малко решение на система от уравнения 36
1.9 Изоморфни области на Скот 44

2 Езикът `REG` **49**
2.1 Регулярни изрази 49
2.2 Операционна семантика 50
2.3 Денотационна семантика 51
2.4 Теорема за еквивалентност 54
2.5 Разширения 59
2.5.1 Езикът `REG++` 59
2.5.2 Езикът `CFG` 61

3 Езикът `FUN` **64**
3.1 Синтаксис 64
3.2 Денотационна семантика 66
3.2.1 Термални оператори 66
3.2.2 Непрекъснатост на термалните оператори 68
3.3 Операционна семантика 73
3.4 Теорема за еквивалентност 76

4 Езикът `PCF` **83**
4.1 Синтаксис 83
4.2 Добре типизирани термове 87
4.3 Операционна семантика 91

4.4 Денотационна семантика 98
4.5 Един важен пример 106
4.6 Коректност 108
4.7 Адекватност 111
4.8 Контекстна еквивалентност 119
4.9 Езикът `PCF(bool)` 129
4.9.1 Типизираща релация 129
4.9.2 Операционна семантика 129
4.9.3 Денотационна семантика 130
4.10 Проблемът за пълна абстракция 131
4.10.1 Езикът `PCF(por)` 135

5 Езикът `LAMBDA` **136**
5.1 Операционна семантика 137
5.2 Кодиране на аритметиката 138
5.2.1 Естествени числа 139
5.2.2 Събиране 139
5.2.3 Умножение 139
5.2.4 Степенуване 140
5.2.5 Наредени двойки 141
5.2.6 Изваждане 142
5.3 Денотационна семантика 143
5.4 Рекурсия 147

Глава 1

Теоретични основи

[4, Глава 5]

В тази глава ще разгледаме понятията, които са ни нужни за дефинирането на денотационна семантика на една програма.

1.1 Частични наредби

Бинарната релация $\sqsubseteq$ върху множеството A се нарича **частична наредба**, ако тя е:

На англ. *partial order*

- рефлексивна, т.е. $(\forall a \in A)[a \sqsubseteq a]$;
- транзитивна, т.е. $(\forall a, b, c \in A)[a \sqsubseteq b \;\&\; b \sqsubseteq c \implies a \sqsubseteq c]$;
- антисиметрична, т.е. $(\forall a, b \in A)[a \sqsubseteq b \;\&\; b \sqsubseteq a \implies a = b]$.

Една такава двойка $(A, \sqsubseteq)$ се нарича частично наредено множество.

Пример 1.1. Да означим

$$\mathcal{F}_n \stackrel{\text{деф}}{=} \{f : \mathbb{N}^n \to \mathbb{N} \mid f \text{ е частична функция}\}.$$

Дефинираме и релацията **включване** между две частични функции по следния начин:

Често вместо $x_1, \dots, x_n$ ще пишем просто $\overline{x}$.

$$f \subseteq g \stackrel{\text{деф}}{\equiv} (\forall \overline{x} \in \mathbb{N}^n)[f(\overline{x}) \text{ не е деф.} \vee (f(\overline{x}) \text{ е деф.} \;\&\; g(\overline{x}) \text{ е деф.} \;\&\; f(\overline{x}) = g(\overline{x}))].$$

Да дефинираме също **графиката** на частичната функция f като

$$\texttt{Graph}(f) \stackrel{\text{деф}}{=} \{\langle \overline{x}, y\rangle \mid f(\overline{x}) = y\}.$$

Тогава лесно се съобразява, че

$$f \subseteq g \iff \texttt{Graph}(f) \subseteq \texttt{Graph}(g).$$

Съобразете, че двойката $(\mathcal{F}_n, \subseteq)$ е частично наредено множество.

Казваме, че a_0 е **най-малък елемент** на частично нареденото множество $(A, \sqsubseteq)$, ако $(\forall a \in A)[a_0 \sqsubseteq a]$. Ако такъв елемент съществува, то той е единствен, защото релацията $\sqsubseteq$ е антисиметрична. За по-кратко монотонно-растящите редици от елементи на A,

Обърнете внимание, че има и понятие *минимален* елемент, което в общия случай е различно от понятието *най-малък* елемент. Един елемент a_0 е минимален за множеството A, ако $\neg(\exists a \in A)[a \sqsubseteq a_0 \;\&\; a \neq a_0]$. Съобразете, че е възможно едно частично наредено множество да притежава повече от един минимални елементи.

$$a_0 \sqsubseteq a_1 \sqsubseteq \cdots \sqsubseteq a_n \sqsubseteq \cdots,$$

ще наричаме (растящи) **вериги**.

Един елемент b е **горна граница** на веригата $(a_n)_{n=0}^{\infty}$, ако $(\forall n)[a_n \sqsubseteq b]$. Един елемент b е **точна горна граница** на веригата $(a_n)_{n=0}^{\infty}$, ако са изпълнени свойствата:

- $(\forall n)[a_n \sqsubseteq b]$, т.е. b е горна граница;
- за всяка друга горна граница c е изпълнено, че $b \sqsubseteq c$, т.е. b е най-малкият елемент измежду всички горни граници на веригата $(a_n)_{n=0}^{\infty}$.

Не всяка верига притежава точна горна граница. Обикновено точната горна граница на вергата $(a_n)_{n=0}^{\infty}$ ще бележим като $\bigsqcup_n a_n$.

Определение 1.1. Наредена тройка от вида $\mathcal{A} = (A, \sqsubseteq, \bot)$ се нарича **област на Скот**, ако:

- $\sqsubseteq$ е бинарна релация върху A, която задава частична наредба.
- Всяка растяща верига $(a_n)_{n=0}^{\infty}$ в A притежава точна горна граница $\bigsqcup_n a_n$.
- $\bot \in A$ е най-малкият елемент на A;

На англ. *Scott domain*. Обикновено в литературата, за да се нарече едно частично нареденото множество област на Скот се изискват още допълнителни свойства, но за нашите цели тази дефиниция ще свърши работа.

Интуицията зад израза $a \sqsubseteq b$ е, че b носи повече информация от a, без да противоречи на a. Елементът $\bot$ означава липса на информация.

Пример 1.2. Тройката $[\mathbb{N}^n \overset{ч}{\rightarrow} \mathbb{N}] \overset{деф}{=} (\,\mathcal{F}_n,\ \subseteq,\ \emptyset^{(n)}\,)$ е област на Скот, където:

- С $\mathcal{F}_n$ означаваме всички частични функции от $\mathbb{N}^n$ в $\mathbb{N}$.
- релацията „включване” между функции е дефинирана по следния начин:

$$f \subseteq g \overset{деф}{\equiv} \mathsf{Graph}(f) \subseteq \mathsf{Graph}(g).$$

- $\emptyset^{(n)}$ е функцията с празна дефиниционна област, т.е. $\mathtt{dom}(\emptyset^{(n)}) = \emptyset$.

В хаскел $\bot$ се означава като `undefined`. Повече за денотационна семантика в хаскел може да прочетете тук.

Пример 1.3. Да разгледаме няколко примера, които вече сме срещали.

- $(\mathscr{P}(\mathbb{N}), \subseteq, \emptyset)$ е област на Скот.
- $(\mathbb{N}, \leq, 0)$ не е е област на Скот.
- $(\mathbb{N} \cup \{\infty\}, \leq, 0)$ е област на Скот, където наредбата $\leq$ е зададена като

$$0 \leq 1 \leq \cdots \leq \infty.$$

- $(\{0,1\}^\star, \preceq, \varepsilon)$ не е област на Скот, където $\preceq$ е релацията префикс на две думи.

Пример 1.4. Да разгледаме множеството

$$Bin^\infty = \{\sigma \mid \sigma : \{0,1,2,\dots,n-1\} \to \{0,1\} \ \&\ n \in \mathbb{N}\} \cup \{f \mid f : \mathbb{N} \to \{0,1\}\}$$

съставено от всички крайни и безкрайни двоични низове.

- Да разгледаме релацията

$$\sigma \preceq \tau \iff |\sigma| \leq |\tau| \ \&\ (\forall i < |\sigma|)[\sigma(i) = \tau(i)],$$

т.е. σ е префикс на τ.

- Да означим с ε единствения двоичен низ с дължина 0. С други думи, ε е празната функция.

Тогава $Bin^\infty = (Bin^\infty, \preceq, \varepsilon)$ е област на Скот.

Тези две свойства ще се окажат полезни по-нататък.

Задача 1.1. Нека $(a_\ell)_{\ell=0}^\infty$ и $(b_k)_{k=0}^\infty$ са вериги в областта на Скот $\mathcal{A}$, за които е изпълнено, че $a_i \sqsubseteq b_i$ за всеки индекс i. Докажете, че $\bigsqcup_\ell a_\ell \sqsubseteq \bigsqcup_k b_k$.

Доказателство. Понеже $\bigsqcup_k b_k$ е точна горна граница на веригата $(b_k)_{k=0}^\infty$, то за всяко k, $b_k \sqsubseteq \bigsqcup_k b_k$. Понеже също $a_i \sqsubseteq b_i$ за всяко i и релацията $\sqsubseteq$ е транзитивна, получаваме, че за всяко i, $a_i \sqsubseteq \bigsqcup_k b_k$. Това означава, че $\bigsqcup_k b_k$ е *горна граница* на веригата $(a_\ell)_{\ell=0}^\infty$ и оттук заключаваме, че $\bigsqcup_\ell a_\ell \sqsubseteq \bigsqcup_k b_k$. $\square$

Задача 1.2. Нека $(a_i)_{i=0}^\infty$ е верига в областта на Скот $\mathcal{A}$ и нека k е естествено число. Докажете, че $\bigsqcup_i a_i = \bigsqcup_i a_{i+k}$.

Езикът $\{a^n b^n \mid n \in \mathbb{N}\}$ може да се представи като обединение на безкрайна верига от крайни езици.

Задача 1.3. Нека Σ е азбука. Да разгледаме тройката $\mathcal{R} = (Reg, \subseteq, \emptyset)$, където Reg е съвкупността от всички регулярни езици над Σ. Вярно ли е, че $\mathcal{R}$ е област на Скот ?

Задача 1.4. Нека $\mathcal{P} = (P, \preceq)$ да бъде частична наредба. Да дефинираме частичната наредба $\mathcal{C} = (\mathtt{Chain}(P), \sqsubseteq)$, където:

- $\mathtt{Chain}(P) = \{\overline{x} \mid \overline{x} = (x_i)_{i=0}^\infty$ е верига в $P\}$;
- $\overline{x} \sqsubseteq \overline{x}' \iff (\forall i)[\, x_i \preceq x_i' \,]$

Докажете, че ако $\mathcal{P}$ формира област на Скот, то $\mathcal{C}$ също формира област на Скот.

1.2 Конструкции на области на Скот

Ще разгледаме няколко конструкции, с които ще видим как можем да строим все по-сложни области на Скот.

1.2.1 Плоска област на Скот

Ще започнем с една проста конструкция, чрез която можем да разширим всяко множество до област на Скот по един почти тривиален начин.

Да фиксираме едно произволно непразно множество A и един елемент $\perp \notin A$. Да означим $A_\perp = A \cup \{\perp\}$ и да разгледаме следната бинарна релация $\sqsubseteq$ върху $A_\perp$:

$$a \sqsubseteq b \iff a = \perp \lor a = b.$$

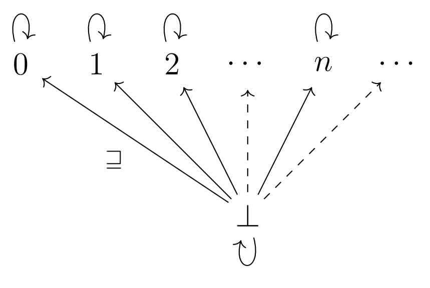
Лесно се съобразява, че $\sqsubseteq$ задава *частична наредба* върху $A_\perp$:

От деф. на $\sqsubseteq$ следва, че $\perp$ е най-малкият елемент

- *рефлексивност*: $a \sqsubseteq a$ за всяка $a \in A_\perp$;
- *транзитивност*: $a \sqsubseteq b \;\&\; b \sqsubseteq c \implies a \sqsubseteq c$ за всеки $a, b, c \in A_\perp$;
- *антисиметричност*: $a \sqsubseteq b \;\&\; b \sqsubseteq a \implies a = b$ за всеки $a, b \in A_\perp$.

Наредбата $(A_\perp, \sqsubseteq)$ ще наричаме **плоска наредба**. Тя ще играе важна роля в нашите разглеждания. Например, често ще разглеждаме плоската наредба $(\mathbb{N}_\perp, \sqsubseteq)$.

$\perp$ се нарича *bottom* елемент



Фигура 1.1: Графично представяне на плоската наредба $\sqsubseteq$ върху $\mathbb{N}_\perp$

Твърдение 1.1. Нека A е произволно множество и нека елементът $\perp \notin A$. Определяме наредената тройка $\mathcal{A}_\perp = (A_\perp, \sqsubseteq, \perp)$ като:

На англ. *flat domain*

- $A_\perp = A \cup \{\perp\}$;
- $\sqsubseteq$ задава *плоската наредба* върху $A_\perp$.

Тогава $\mathcal{A}_\perp$ е област на Скот, която ще наричаме **плоска област на Скот** за множеството A.

1.2.2 Крайно произведение

Това е една от най-простите конструкции върху области на Скот. Вижте например [12, стр. 125] и [4, с. 176].

Нека $\mathcal{A}$ и $\mathcal{B}$ са области на Скот. Тогава $\mathcal{A} \times \mathcal{B} \stackrel{\text{деф}}{=} (A \times B, \sqsubseteq, \perp)$, където

- $A \times B \stackrel{\text{деф}}{=} \{\langle a, b\rangle \mid a \in A \,\&\, b \in B\}$;
- $\langle a, b\rangle \sqsubseteq \langle a', b'\rangle \iff a \sqsubseteq^{\mathcal{A}} a' \,\&\, b \sqsubseteq^{\mathcal{B}} b'$;
- $\perp \stackrel{\text{деф}}{=} \langle \perp^{\mathcal{A}}, \perp^{\mathcal{B}}\rangle$.

Твърдение 1.2. Ако $\mathcal{A}$ и $\mathcal{B}$ са области на Скот, то $\mathcal{A} \times \mathcal{B}$ е област на Скот.

Упътване. Лесно се съобразява, че $\sqsubseteq$ е частична наредба и че $\perp$ е най-малкият елемент. Да разгледаме една верига $\{\langle a_i, b_i\rangle\}_{i=1}^{\infty}$ в $\mathcal{A} \times \mathcal{B}$. Лесно се вижда, че

$$\bigsqcup_i \langle a_i, b_i\rangle = \langle \bigsqcup_i a_i, \bigsqcup_i b_i\rangle.$$

- За произволен елемент $\langle a_i, b_i\rangle$ от веригата е ясно, че $a_i \sqsubseteq^{\mathcal{A}} \bigsqcup_i a_i$ и $b_i \sqsubseteq^{\mathcal{B}} \bigsqcup_i b_i$. Следователно, $\langle \bigsqcup_i a_i, \bigsqcup_i b_i\rangle$ е горна граница на веригата.
- Нека $\langle c, d\rangle$ е горна граница на веригата, т.е. за всяко i, $a_i \sqsubseteq^{\mathcal{A}} c$ и $b_i \sqsubseteq^{\mathcal{B}} d$. Но тогава c е горна граница на веригата $(a_i)_{i=0}^{\infty}$ и следователно $\bigsqcup_i a_i \sqsubseteq^{\mathcal{A}} c$. Също така, d е горна граница на веригата $(b_i)_{i=0}^{\infty}$ и следователно $\bigsqcup_i b_i \sqsubseteq^{\mathcal{B}} d$. Заключаваме, че $\langle \bigsqcup_i a_i, \bigsqcup_i b_i\rangle \sqsubseteq \langle c, d\rangle$, т.е. $\langle \bigsqcup_i a_i, \bigsqcup_i b_i\rangle$ е точна горна граница на веригата.

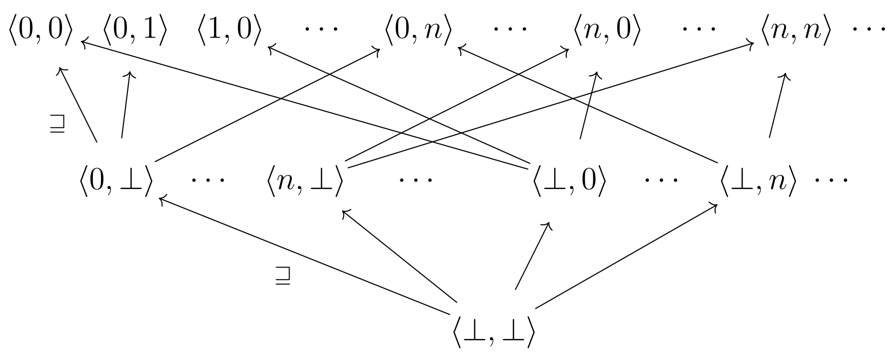
$\square$

Нека $\mathcal{A}_1, \ldots, \mathcal{A}_n$, $n \geq 2$, са области на Скот. Дефинираме $\prod_{i=1}^{n} \mathcal{A}_i = (A, \sqsubseteq, \perp)$ по следния начин:

- Ако $n = 2$, то $\prod_{i=1}^{2} \mathcal{A}_i \stackrel{\text{деф}}{=} \mathcal{A}_1 \times \mathcal{A}_2$.
- Ако $n > 2$, то $\prod_{i=1}^{n} \mathcal{A}_i \stackrel{\text{деф}}{=} (\prod_{i=1}^{n-1} \mathcal{A}_i) \times \mathcal{A}_n$.

Използвайки *Твърдение* 1.2, лесно се съобразява следното твърдение.

Твърдение 1.3. Ако $\mathcal{A}_1, \ldots, \mathcal{A}_n$, $n \geq 2$, са области на Скот, то $\prod_{i=1}^{n} \mathcal{A}_i$ е област на Скот.



Фигура 1.2: Графично представяне на част от $\sqsubseteq$ върху $\mathbb{N}_\perp^2$

Вижда се от *Фигура* 1.2, че всяка верига в $\mathbb{N}_\perp^2$ има дължина най-много 3. Лесно се съобразява, че всяка верига в $\mathbb{N}_\perp^k$ има дължина най-много $k+1$. Свойството, че всяка верига в $\mathbb{N}_\perp^k$ има само краен брой различни члена ще се окаже важно по-нататък. Сега ще въведем понятие, което описва това свойство в произволна област на Скот.

$\mathbb{N}_\perp^k \stackrel{\text{деф}}{=} \underbrace{\mathbb{N}_\perp \times \cdots \times \mathbb{N}_\perp}_{k}$. Какво става, когато $k = 0$? Това ще бъде важно по-нататък.

Нека $\mathcal{A}$ е област на Скот и да разгледаме една верига $(a_n)_{n=0}^\infty$ в $\mathcal{A}$. Ще казваме, че $(a_n)_{n=0}^\infty$ се **стабилизира**, ако съществува индекс n_0, за който

$$(\forall n \geq n_0)[a_{n_0} = a_n],$$

т.е.

$$a_0 \sqsubseteq a_1 \sqsubseteq a_2 \sqsubseteq \cdots \sqsubseteq a_{n_0} = a_{n_0+1} = a_{n_0+2} = \cdots$$

От казаното по-горе следва, че всяка растяща верига в $\mathbb{N}_\perp^k$ се стабилизира.

Едни от основните области на Скот, които ще разглеждаме при дефинирането на денотационната семантика ще бъдат $\mathbb{N}_\perp$ и $\mathbb{N}_\perp^k$.

1.3 Изображения в области на Скот

Нека $\mathcal{A}_i = (A_i,\ \sqsubseteq_i,\ \bot_i)$, за $i = 1, 2$, са области на Скот. Ще въведем няколко основни видове изображения между $\mathcal{A}_1$ и $\mathcal{A}_2$, които ще използваме често. След това ще разгледаме свойства на тези изображения и ще видим каква е връзката между тях.

- Всяка тотална функция от вида $f : A_1 \to A_2$ ще наричаме изображение между областите на Скот $\mathcal{A}_1$ и $\mathcal{A}_2$ и ще записваме $f : \mathcal{A}_1 \to \mathcal{A}_2$. Да въведем означението

Тук $\bot$ е различно от $\perp$, макар и трудно да се различават графически. Освен това, $\mathcal{A}$ е различно от A.

$$[\mathcal{A}_1 \ \to\ \mathcal{A}_2] \overset{\text{деф}}{=} (\{f \mid f : \mathcal{A}_1 \to \mathcal{A}_2\},\ \sqsubseteq,\ \bot),$$

където имаме следната релация между изображенията $f, g : \mathcal{A}_1 \to \mathcal{A}_2$:

✍ Сами проверете, че така дефинираната релация $\sqsubseteq$ задава частична наредба, при положение, че $\sqsubseteq_2$ е частична наредба.

$$f \sqsubseteq g \overset{\text{деф}}{\equiv} (\forall a \in \mathcal{A}_1)[f(a) \sqsubseteq_2 g(a)].$$

Също така, изображението $\bot : \mathcal{A}_1 \to \mathcal{A}_2$ е дефинирано като

$$(\forall a \in \mathcal{A}_1)[\bot(a) = \bot_2].$$

На хаскел можем да дефинираме изображението $\bot$ по следния начин:

```
ghci> let bottom _ = undefined
```

Следващата теорема е много важна, защото тя ни показва основният метод, с който можем да строим по-сложни и все по-сложни области на Скот.

Теорема 1.1. Ако $\mathcal{A}$ и $\mathcal{B}$ са области на Скот, то $[\mathcal{A} \ \to\ \mathcal{B}]$ също е област на Скот.

Доказателство. Нетривиалната част в доказателството е да проверим, че всяка верига $(f_i)_{i=0}^{\infty}$ в $[\mathcal{A} \ \to\ \mathcal{B}]$ притежава точна горна граница. Да разгледаме изображението $h : \mathcal{A} \to \mathcal{B}$, където:

$$h(a) \overset{\text{деф}}{=} \bigsqcup\{f_i(a) \mid i \in \mathbb{N}\}. \tag{1.1}$$

Ще докажем, че h е тази точна горна граница.

- Първо, трябва да се убедим, че дефиницията на h е „смислена”, т.е. h е тотална функция. Трябва да докажем, че за всяко $a \in \mathcal{A}$, точната горна граница

Това задължително трябва да се провери, защото например множеството $\{\bot, 0, 3\}$ *няма* точна горна граница относно плоската наредба в $\mathbb{N}_\bot$.

$$\bigsqcup\{f_i(a) \mid i \in \mathbb{N}\}$$

съществува. Да фиксираме произволен елемент $a \in \mathcal{A}$. Получаваме следната верига в $\mathcal{B}$:

$$f_0(a) \sqsubseteq f_1(a) \sqsubseteq f_2(a) \sqsubseteq \cdots$$

Понеже $\mathcal{B}$ е област на Скот, то тази верига притежава точна горна граница в $\mathcal{B}$, която означачаваме като $\bigsqcup\{f_i(a) \mid i \in \mathbb{N}\}$, а според нашата дефиниция, това е точно $h(a)$. Това означава, че h е тотална функция.

- Дотук имаме, че $h \in [\mathcal{A} \to \mathcal{B}]$. Лесно се събразява, че h е горна граница на веригата $(f_i)_{i=0}^{\infty}$, защото за всеки елемент $a \in \mathcal{A}$ и произволен индекс k,
$$f_k(a) \sqsubseteq \bigsqcup\{f_i(a)|i \in \mathbb{N}\} \stackrel{\text{деф}}{=} h(a).$$
- Сега остава да проверим, че h е точна горна граница, т.е. h е най-малката измежду всички горни граници на веригата $(f_i)_{i=0}^{\infty}$. Нека g е друга горна граница на $(f_i)_{i=0}^{\infty}$. Това означава, че за всеки индекс i, $f_i \sqsubseteq g$. Следователно, за фиксирано $a \in \mathcal{A}$, $g(a)$ е горна граница за веригата $(f_i(a))_{i=0}^{\infty}$. Тогава е ясно, че за разглеждания елемент a,
$$h(a) \stackrel{\text{деф}}{=} \bigsqcup\{f_i(a) \mid i \in \mathbb{N}\} \sqsubseteq g(a).$$
Понеже елементът a е прозиволен, получаваме, че $h \sqsubseteq g$.
- Доказахме, че h е горна граница и че h е най-малката измежду всички горни граници. Заключваме, че h е *точната горна граница* на веригата $(f_i)_{i=0}^{\infty}$. С други думи,
$$h = \bigsqcup_i f_i.$$

Получаваме, че
$$(\bigsqcup_i f_i)(a) = \bigsqcup_i\{f_i(a)\}.$$
Добре е да свикнете с тези означения, защото понататък ще ги използваме често.

$\square$

Завършваме с един важен частен случай на горната теорема.

Следствие 1.1. $[\mathbb{N}_{\perp}^n \to \mathbb{N}_{\perp}]$ е област на Скот.

1.3.1 Монотонни изображения

Да разгледаме областите на Скот $\mathcal{A}_1 = (A_1, \sqsubseteq_1, \perp_1)$ и $\mathcal{A}_2 = (A_2, \sqsubseteq_2, \perp_2)$.

Определение 1.2. Едно изображение $f : \mathcal{A}_1 \to \mathcal{A}_2$ се нарича **монотонно**, ако
$$(\forall a \in \mathcal{A}_1)(\forall a' \in \mathcal{A}_1)[a \sqsubseteq_1 a' \implies f(a) \sqsubseteq_2 f(a')].$$

Да въведем означението

$$[\mathcal{A}_1 \stackrel{\text{м}}{\to} \mathcal{A}_2] \stackrel{\text{деф}}{=} (\{f : \mathcal{A}_1 \to \mathcal{A}_2 \mid f \text{ е мон. изобр.}\},\ \sqsubseteq,\ \bot).$$

Теорема 1.2. Ако $\mathcal{A}_1$ и $\mathcal{A}_2$ са области на Скот, то $[\mathcal{A}_1 \stackrel{\text{м}}{\to} \mathcal{A}_2]$ също е област на Скот.

Упътване. Да фиксираме една верига $(f_i)_{i=0}^{\infty}$ в $[\mathcal{A}_1 \stackrel{\text{м}}{\to} \mathcal{A}_2]$. Трябва да докажем, че тази верига притежава точна горна граница, която е монотонно изображение. Да разгледаме същото изображение $h : \mathcal{A}_1 \to \mathcal{A}_2$ както в доказателството на *Теорема* 1.1, като

$$h(a) \stackrel{\text{деф}}{=} \bigsqcup\{f_i(a) \mid i \in \mathbb{N}\}.$$

Оттам знаем, че h е точна горна граница на веригата. Остава да докажем, че $h \in [\mathcal{A}_1 \stackrel{\text{м}}{\to} \mathcal{A}_2]$. Нека $a \sqsubseteq b$. Тогава, за всеки индекс i, понеже f_i са монотонни изображения, получаваме, че за всеки индекс i, $f_i(a) \sqsubseteq f_i(b)$. Сега можем да приложим *Задача* 1.1 за веригите $(f_i(a))_{i=0}^{\infty}$ и $(f_i(b))_{i=0}^{\infty}$, откъдето получаваме, че

$$h(a) = \bigsqcup_i f_i(a) \sqsubseteq \bigsqcup_i f_i(b) = h(b).$$

□

Оттук като частен случай получаваме следното полезно следствие.

Следствие 1.2. $[\mathbb{N}_\bot^n \stackrel{\text{м}}{\to} \mathbb{N}_\bot]$ е област на Скот.

Удобно е да имаме следващото свойство на монотонните изображения като отделно твърдение за да можем да се позоваваме на него, когато имаме нужда от него по-късно.

Твърдение 1.4. Нека $f \in [\mathcal{A} \stackrel{\text{м}}{\to} \mathcal{B}]$ и $(a_i)_{i=0}^{\infty}$ е верига от елементи на $\mathcal{A}$. Тогава

$$\bigsqcup_i f(a_i) \sqsubseteq f(\bigsqcup_i a_i).$$

Доказателство. Понеже f е монотонно, то $(f(a_i))_{i=0}^{\infty}$ е верига от елементи на областта на Скот $\mathcal{B}$ и тя има точна горна граница. Достатъчно е да докажем, че $f(\bigsqcup_i a_i)$ е горна граница на веригата $(f(a_i))_{i=0}^{\infty}$. Но това е лесно. Понеже за произволен индекс k, $a_k \sqsubseteq \bigsqcup_i a_i$ и f е монотонно изображение, то веднага получаваме, че $f(a_k) \sqsubseteq f(\bigsqcup_i a_i)$. Заключаваме, че $\bigsqcup_k f(a_k) \sqsubseteq f(\bigsqcup_i a_i)$. □

1.3.2 Непрекъснати изображения

Едно изображение $f : \mathcal{A}_1 \to \mathcal{A}_2$ се нарича **непрекъснато**, ако са изпълнени свойствата:

На англ. *continuous*

- f е монотонно изображение;
- при всеки избор на верига $(a_i)_{i=0}^{\infty}$ в $\mathcal{A}_1$, имаме равенството

$$f(\bigsqcup_i a_i) = \bigsqcup\{f(a_i) \mid i \in \mathbb{N}\}.$$

Понеже $\mathcal{A}_1$ е област на Скот знаем, че $\bigsqcup_i a_i \in \mathcal{A}_1$

Понеже f е монотонно, то $(f(a_i))_{i=0}^{\infty}$ е верига.

$\bigsqcup_i f(a_i) \stackrel{\text{деф}}{=} \bigsqcup\{f(a_i) \mid i \in \mathbb{N}\}$

Да означим

$$[\mathcal{A}_1 \stackrel{\text{н}}{\to} \mathcal{A}_2] \stackrel{\text{деф}}{=} (\{f : \mathcal{A}_1 \to \mathcal{A}_2 \mid f \text{ е непр. изобр.}\},\ \sqsubseteq,\ \bot).$$

Ясно е, че всяко непрекъснато изображение е монотонно. Ествено е да си зададем въпроса дали имаме и обратното включване. Оказва се, че в общия случай не е вярно, че всяко монотонно изображение е непрекъснато.

Твърдение 1.5. Съществува област на Скот $\mathcal{A}$, за която

$$[\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{A}] \subsetneqq [\mathcal{A} \stackrel{\text{м}}{\to} \mathcal{A}].$$

Упътване. Нека $A = \{a_n \mid n \in \mathbb{N}\} \cup \{b, c\}$. Да разгледаме областта на Скот $\mathcal{A} = (A, \sqsubseteq, a_0)$, където наредбата между елементите е следната:

За да не претрупваме фигурата, изпускаме да пишем примки на всеки възел, а също и транцитивното затваряне на релацията.

$$\begin{tikzcd}
c \\
b \arrow[u] \\
a_n \arrow[u, dashed] \\
a_2 \arrow[u, dashed] \\
a_1 \arrow[u] \\
a_0 \arrow[u]
\end{tikzcd}$$

Фигура 1.3: Графично представяне на наредбата $\sqsubseteq$ върху $\mathcal{A}$

С други думи, веригата $(a_i)_{i=0}^{\infty}$ има горни граници b и c, като b е точната горна граница.

Нека дефинираме изображението f като положим $f(a_n) = a_{n+1}$, $f(b) = c$ и $f(c) = c$. Очевидно е, че f е монотонно изображение. Лесно се вижда, че f не е непрекъснато изображение, защото

$$f(\bigsqcup_n a_n) = f(b) = c,$$

но от равенствата

$$\bigsqcup_n f(a_n) = \bigsqcup_n a_{n+1} = b$$

получаваме, че

$$f(\bigsqcup_n a_n) \neq \bigsqcup_n f(a_n).$$

□

Сега да видим един важен за нас случай, при който имаме и обратното включване.

В Раздел 1.2.2 дефинирахме какво означава една верига $(n_a)_{a=0}^{\infty}$ да се стабилизира: съществува индекс n_0, за който

$$(\forall n \geq n_0)[a_{n_0} = a_n].$$

Твърдение 1.6. Нека $\mathcal{A}_1$ и $\mathcal{A}_2$ са области на Скот, като всяка верига в $\mathcal{A}_1$ се *стабилизира*. Тогава

$$[\mathcal{A}_1 \overset{\text{м}}{\to} \mathcal{A}_2] \subseteq [\mathcal{A}_1 \overset{\text{н}}{\to} \mathcal{A}_2].$$

Упътване. Да разгледаме една верига $(a_i)_{i=0}^{\infty}$ в $\mathcal{A}_1$ и $f \in [\mathcal{A}_1 \overset{\text{м}}{\to} \mathcal{A}_2]$. Ще докажем, че

$$f(\bigsqcup_i a_i) = \bigsqcup_i f(a_i).$$

Ще използваме свойството, че веригата $(a_i)_{i=0}^{\infty}$ се стабилизира. Нека n_0 е индекс, такъв че $(\forall k \geq n_0)[a_k = a_{n_0}]$. Това означава, че $\bigsqcup_i a_i = a_{n_0}$. Тогава $f(\bigsqcup_i a_i) = f(a_{n_0})$. От друга страна, понеже f е монотонно изображение, то $(f(a_i))_{i=0}^{\infty}$ е също е верига и за фиксирания индекс n_0, $(\forall k \geq n_0)[f(a_k) = f(a_{n_0})]$. Оттук получаваме, че $\bigsqcup_i f(a_i) = f(a_{n_0})$. Заключаваме, че $f(\bigsqcup_i a_i) = \bigsqcup_i f(a_i)$.

□

От Раздел 1.2.2 знаем, че всяка верига в $\mathbb{N}_\perp^n$ се *стабилизира*, то получаваме следното важно следствие.

Следствие 1.3. $[\mathbb{N}_\perp^n \overset{\text{м}}{\to} \mathbb{N}_\perp] = [\mathbb{N}_\perp^n \overset{\text{н}}{\to} \mathbb{N}_\perp]$.

Понеже от *Теорема 1.2* имаме, че монотонните изображения образуват област на Скот, то директно получаваме следната важна теорема.

Теорема 1.3. $[\mathbb{N}^n_\perp \overset{\text{н}}{\to} \mathbb{N}_\perp]$ е област на Скот.

[11, стр. 124]

Задача 1.5. Нека $f \in [\mathcal{A} \overset{\text{м}}{\to} \mathcal{B}]$ и $g \in [\mathcal{B} \overset{\text{м}}{\to} \mathcal{A}]$ имат свойствата:

- $f \circ g = id_{\mathcal{B}}$;
- $g \circ f = id_{\mathcal{A}}$.

Доказжете, че f и g са непрекъснати.

1.4 Област на Скот от непрекъснати изображения

Тук ще докажем, че всички непрекъснати изображения между две области на Скот също образуват област на Скот. За да направим това, първо трябва да се подготвим с една теорема, която е важна, защото с нейна помощ се доказват много свойства на непрекъснатите изображения от по-висок ред.

[12, стр. 127]

[4, Switch Lemma, стр. 178]

Лема 1.1 (Лема за разместването). Нека $\mathcal{A} = (A, \sqsubseteq, \perp)$ да бъде област на Скот и нека множеството

$$\{a_{m,n} \mid m, n \in \mathbb{N}\}$$

от елементи на A притежава свойството, че

$$n \leq n' \,\&\, m \leq m' \;\Rightarrow\; a_{n,m} \sqsubseteq a_{n',m'}.$$

Тогава са изпълнени равенствата

$$\bigsqcup_m (\bigsqcup_n a_{n,m}) = \bigsqcup_n (\bigsqcup_m a_{n,m}) = \bigsqcup_n a_{n,n}.$$

Доказателство. Първо ще въведем някои означения.

- Да фиксираме произволно m. Тогава множеството $\{a_{n,m} \mid n \in \mathbb{N}\}$ образува верига:
$$a_{0,m} \sqsubseteq a_{1,m} \sqsubseteq a_{2,m} \sqsubseteq \cdots$$
Следователно тя има точна горна граница $b_m \stackrel{\text{деф}}{=} \bigsqcup\{a_{n,m} \mid n \in \mathbb{N}\}$.

По дефиниция, всяка монотонно растяща редица в област на Скот притежава точна горна граница.

- Аналогично, при фиксирано n, множеството $\{a_{n,m} \mid m \in \mathbb{N}\}$ образува верига:
$$a_{n,0} \sqsubseteq a_{n,1} \sqsubseteq a_{n,2} \sqsubseteq \ldots,$$
която притежава точна горна граница $c_n \stackrel{\text{деф}}{=} \bigsqcup\{a_{n,m} \mid m \in \mathbb{N}\}$.

Това означава, че трябва да докажем следното:

$$\bigsqcup_m b_m = \bigsqcup_n c_n = \bigsqcup_n a_{n,n}.$$

1) Първо да съобразим, че множеството $\{b_m \mid m \in \mathbb{N}\}$ образува верига в $\mathcal{A}$ и следователно притежава точна горна граница $\bigsqcup_m b_m$. Нека да разгледаме произволни $m \leq m'$. Тогава
$$(\forall n)[a_{n,m} \sqsubseteq a_{n,m'} \sqsubseteq \bigsqcup_k a_{k,m'} = b_{m'}].$$

Следователно $b_{m'}$ е горна граница на веригата $(a_{n,m})_{n=0}^{\infty}$ и понеже b_m е точна горна граница на $(a_{n,m})_{n=0}^{\infty}$, то получаваме, че

$$b_m \sqsubseteq b_{m'}.$$

Това означава, че $(b_m)_{m=0}^{\infty}$ е верига в $\mathcal{A}$ и тя притежава точна горна граница $\bigsqcup_m b_m$.

2) С подобни разсъждения можем да докажем, че множеството $\{c_n \mid n \in \mathbb{N}\}$ образува верига в $\mathcal{A}$, която притежава точна горна граница $\bigsqcup_n c_n$.

3) Сега ще докажем, че

$$\bigsqcup_m b_m = \bigsqcup_n c_n.$$

Имаме, че

$$(\forall m)(\forall n)[a_{n,m} \sqsubseteq \bigsqcup_i a_{i,m} = b_m \sqsubseteq \bigsqcup_i b_i],$$

което е еквивалентно на

$$(\forall n)(\forall m)[a_{n,m} \sqsubseteq b_m \sqsubseteq \bigsqcup_i b_i].$$

Да фиксираме произволно n. Тогава $\bigsqcup_i b_i$ е горна граница на веригата $(a_{n,i})_{i=0}^{\infty}$. Следователно, $c_n = \bigsqcup_i a_{n,i} \sqsubseteq \bigsqcup_i b_i$. Така получаваме, че $\bigsqcup_i b_i$ е горна граница и на веригата $(c_n)_{n=0}^{\infty}$ и тогава

$$\bigsqcup_n c_n \sqsubseteq \bigsqcup_i b_i.$$

С аналогични разсъждения можем да докажем също, че

$$\bigsqcup_m b_m \sqsubseteq \bigsqcup_n c_n.$$

Така доказахме, че

$$\bigsqcup_m b_m = \bigsqcup_n c_n.$$

4) Достатъчно е още да докажем, че

$$\bigsqcup_n a_{n,n} = \bigsqcup_n c_n.$$

Ясно е, че $a_{n,n}$ е елемент на веригата $(a_{n,m})_{m=0}^{\infty}$ и следователно $a_{n,n} \sqsubseteq \bigsqcup_m a_{n,m} = c_n \sqsubseteq \bigsqcup_n c_n$. Получаваме, че $\bigsqcup_n c_n$ е горна граница на веригата $(a_{n,n})_{n=0}^{\infty}$ и следователно $\bigsqcup_n a_{n,n} \sqsubseteq \bigsqcup_n c_n$.

За другата посока, да разгледаме произволен елемент $a_{n,m}$. Нека $k = \max\{n, m\}$. Ясно е, че $a_{n,m} \sqsubseteq a_{k,k} \sqsubseteq \bigsqcup_n a_{n,n}$. Следователно, $\bigsqcup_n a_{n,n}$ е горна граница на верига $(a_{m,n})_{m=0}^{\infty}$ и оттук получаваме, че за всяко n, $c_n \sqsubseteq \bigsqcup_n a_{n,n}$. Получаваме, че $\bigsqcup_n a_{n,n}$ е горна граница на веригата $(c_n)_{n=0}^{\infty}$ и следователно $\bigsqcup_n c_n \sqsubseteq \bigsqcup_n a_{n,n}$.

С това доказателството на теоремата е завършено. □

Лема 1.2. Нека $\mathcal{A}$ и $\mathcal{B}$ са области на Скот. Нека $(f_k)_{k=0}^{\infty}$ е верига от елементи на $[\mathcal{A} \xrightarrow{\text{н}} \mathcal{B}]$. Да дефинираме изображението h на $\mathcal{A}$ в $\mathcal{B}$ по следния начин:

$$h(a) \stackrel{\text{деф}}{=} \bigsqcup\{f_k(a) \mid k \in \mathbb{N}\}.$$

Изображението h е *непрекъснато* и е *точна горна граница* на веригата $(f_k)_{k=0}^{\infty}$, т.е. $h = \bigsqcup_k f_k$.

Ако $b_k = f_k(a)$, то $h(a)$ е точната горна граница на веригата $(b_k)_{k=0}^{\infty}$ в $\mathcal{B}$

Доказателство. Доказателството, че h е точна горна граница на веригата $(f_k)_{k=0}^{\infty}$ е лесно.

- Да разгледаме произволен елемент $a \in A$. Лесно се вижда, че понеже $(f_k)_{k=0}^{\infty}$ е верига, то $(f_k(a))_{k=0}^{\infty}$ също е верига.

  $\bigsqcup_n f_n(a)$ е съкратен запис за $\bigsqcup\{f_n(a) \mid n \in \mathbb{N}\}$.

  Получаваме, че за всяко k, $f_k(a) \sqsubseteq^{\mathcal{B}} \bigsqcup_n f_n(a) \stackrel{\text{деф}}{=} h(a)$. Понеже това е вярно за произволно $a \in A$, $(\forall k)[f_k \sqsubseteq h]$, което означава, че h е горна граница на веригата.

- Да разгледаме произволно изображение g, което е горна граница на веригата $(f_k)_{k=0}^{\infty}$. За произволен елемент $a \in A$,

  $$(\forall k)[f_k(a) \sqsubseteq^{\mathcal{B}} g(a)].$$

  Това означава, че $g(a)$ е горна граница на веригата $(f_k(a))_{k=0}^{\infty}$. Понеже $h(a) = \bigsqcup_k\{f_k(a)\}$ е точната горна граница на веригата $(f_k(a))_{k=0}^{\infty}$, то $h(a) \sqsubseteq^{\mathcal{B}} g(a)$. Оттук следва, че $h \sqsubseteq g$.

По-сложната част на доказателството е проверката, че h е непрекъснато изображение. Да вземем една монотонно растяща редица $(a_k)_{k=0}^{\infty}$ от елементи на A. Ще докажем, че

За момента дори не е ясно дали $\{h(a_k) \mid k \in \mathbb{N}\}$ е верига в $\mathcal{B}$

$$h(\bigsqcup_k a_k) = \bigsqcup_k\{h(a_k)\}.$$

Нека $e_{n,m} \stackrel{\text{деф}}{=} f_n(a_m)$. Понеже всяко f_n е непрекъснато и следователно монотонно изображение, то имаме

$$n \leq n' \;\&\; m \leq m' \;\Rightarrow\; e_{n,m} \sqsubseteq^{\mathcal{B}} e_{n',m'}.$$

Следователно,

$$h(\bigsqcup_m a_m) = \bigsqcup_n (f_n(\bigsqcup_m a_m)) \qquad \text{// от деф. на } h$$

$$
\begin{aligned}
&= \bigsqcup_n (\bigsqcup_m f_n(a_m)) && \text{// защото } f_n \text{ е непр.}\\
&= \bigsqcup_n (\bigsqcup_m e_{n,m}) = \bigsqcup_m (\bigsqcup_n e_{n,m}) && \text{// от \textit{Лема} 1.1}\\
&= \bigsqcup_m (\bigsqcup_n f_n(a_m)) && \text{// от деф. на } e_{n.m}\\
&= \bigsqcup_m \{h(a_m)\}. && \text{// от деф. на } h
\end{aligned}
$$

□

Да напомним, че релацията $\sqsubseteq$ между две изображения е дефинирана като

$$f \sqsubseteq g \overset{\text{деф}}{\equiv} (\forall a \in A)[f(a) \sqsubseteq^{\mathcal{B}} g(a)].$$

Теорема 1.4. Ако $\mathcal{A}$ и $\mathcal{B}$ са области на Скот, то $[\mathcal{A} \overset{\text{н}}{\to} \mathcal{B}]$ е област на Скот.

Следващото твърдение ни казва, че едно изображение е непрекъснато точно тогава, когато е непрекъснато по всеки от аргументите си.

[4, стр. 184]

Твърдение 1.7. Нека $f \in [\mathcal{A} \times \mathcal{B} \to \mathcal{C}]$. Да положим $f_a(y) \overset{\text{деф}}{=} f(a, y)$, за произволно $a \in \mathcal{A}$ и $f_b(x) \overset{\text{деф}}{=} f(x, b)$, за произволно $b \in \mathcal{B}$. Тогава следните условия са еквивалентни:

(а) $f \in [\mathcal{A} \times \mathcal{B} \overset{\text{н}}{\to} \mathcal{C}]$;

(б) f_a и f_b са непрекъснати за произволни a и b.

Доказателство. ($\Rightarrow$) Лесно се съобразява, че ако f е непрекъснато изображение, то f е непрекъснато по всеки от аргументите си. Да видим например защо f е непрекъснато по първия аргумент. Да разгледаме веригата $(\langle a_i, b\rangle)_{i=0}^{\infty}$ от елементи на $\mathcal{A} \times \mathcal{B}$, за някое фиксирано b. Знаем, че $\bigsqcup_i \langle a_i, b\rangle = \langle \bigsqcup_i a_i, b\rangle$. Тогава

Формално погледнато, правилно е да пишем $f(\langle a, b\rangle)$ вместо $f(a, b)$.

$$
\begin{aligned}
f(\bigsqcup_i a_i, b) &= f(\bigsqcup_i \langle a_i, b\rangle)\\
&= \bigsqcup_i f(a_i, b) && \text{// } f \text{ е непрекъснато.}
\end{aligned}
$$

($\Leftarrow$) Нека сега f е непрекъснато по всеки от аргументите си. Ще докажем, че f е непрекъснато. Нека $\{\langle a_n, b_n\rangle\}_{n=0}^{\infty}$ е верига в $\mathcal{A}_1 \times \mathcal{A}_2$. Понеже от

Твърдение 1.2 знаем, че

$$\bigsqcup_n \langle a_n, b_n \rangle = \langle \bigsqcup_n a_n, \bigsqcup_n b_n \rangle,$$

ще докажем, че

$$\bigsqcup_n f(a_n, b_n) = f(\bigsqcup_n a_n, \bigsqcup_n b_n).$$

Да положим $c_{n,m} \stackrel{\text{деф}}{=} f(a_n, b_m)$. Понеже f е непрекъснато по всеки от аргументите си, лесно се вижда, че f е монотонно изображение по всеки от аргументите си. Следователно,

$$n \leq n' \;\&\; m \leq m' \implies c_{n,m} \sqsubseteq c_{n',m'}.$$

Получаваме, че

$$\begin{aligned}
\bigsqcup_n f(a_n, b_n) &= \bigsqcup_n c_{n,n} && \text{// от опр. на } c_{n,m} \\
&= \bigsqcup_n (\bigsqcup_m c_{n,m}) && \text{// от } \textit{Лема } 1.1 \\
&= \bigsqcup_n (\bigsqcup_m f(a_n, b_m)) && \text{// от опр. на } c_{n,m} \\
&= \bigsqcup_n f(a_n, \bigsqcup_m b_m) && \text{// } f \text{ е непр. по втория си аргумент} \\
&= f(\bigsqcup_n a_n, \bigsqcup_m b_m) && \text{// } f \text{ е непр. по първия си аргумент.}
\end{aligned}$$

□

Това твърдение се обобщава по естествен начин за повече от два аргумента.

Следствие 1.4. Нека $f \in [\mathcal{A}_1 \times \cdots \times \mathcal{A}_n \to \mathcal{B}]$. Тогава следните условия са еквивалентни:

(1) $f \in [\mathcal{A}_1 \times \cdots \times \mathcal{A}_n \stackrel{\text{н}}{\to} \mathcal{B}]$;

(2) за произволно $i = 1, \ldots, n$ и произволни $a_j \in \mathcal{A}_j$ за $i \neq j$, изображението $g \in [\mathcal{A}_i \stackrel{\text{н}}{\to} \mathcal{B}]$, където $g(x) \stackrel{\text{деф}}{=} f(a_1, \ldots, a_{i-1}, x, a_{i+1}, \ldots, a_n)$.

1.5 Основни непрекъснати изображения

Да започнем като първо дефинираме следните *основни изображения*

$$
\begin{aligned}
&\texttt{(+)} : \mathbb{N}_\perp^2 \to \mathbb{N}_\perp, \text{ където}\\
&\texttt{(+)}(a,b) \stackrel{\text{деф}}{=} \begin{cases} a+b, & \text{ако } a,b \in \mathbb{N}\\ \perp, & \text{ако } \perp \in \{a,b\}\end{cases}\\
&\texttt{(-)} : \mathbb{N}_\perp^2 \to \mathbb{N}_\perp, \text{ където}\\
&\texttt{(-)}(a,b) \stackrel{\text{деф}}{=} \begin{cases} 0, & \text{ако } a,b \in \mathbb{N} \,\&\, a < b\\ a-b, & \text{ако } a,b \in \mathbb{N} \,\&\, a \geq b\\ \perp, & \text{ако } \perp \in \{a,b\}\end{cases}\\
&\texttt{(==)} : \mathbb{N}_\perp^2 \to \mathbb{N}_\perp, \text{ където}\\
&\texttt{(==)}(a,b) \stackrel{\text{деф}}{=} \begin{cases} 1, & \text{ако } a = b \,\&\, a,b \in \mathbb{N}\\ 0, & \text{ако } a \neq b \,\&\, a,b \in \mathbb{N}\\ \perp, & \text{ако } \perp \in \{a,b\}\end{cases}
\end{aligned}
$$

Задача 1.6. Докажете, че изображенията `(+)`, `(-)` и `(==)` са непрекъснати.

Упътване. Понеже $[\mathbb{N}_\perp^n \xrightarrow{\text{м}} \mathbb{N}_\perp] = [\mathbb{N}_\perp^n \xrightarrow{\text{н}} \mathbb{N}_\perp]$, достатъчно е да докажете, че изображенията са монотонни. □

Така дефинираните функции `(+)`, `(-)` и `(==)` са ограничени варианти, само до $\mathbb{N}_\perp$, на едноименните функции в хаскел. Единствено може би има нужда да се убедим, че и на хаскел не можем да сравяваме с $\perp$.

```
ghci> :t (==)
(==) :: Eq a => a -> a -> Bool
ghci> (==) 2 3
False
ghci> (==) 2 undefined
*** Exception: Prelude.undefined
ghci> (==) undefined 2
*** Exception: Prelude.undefined
ghci> (==) undefined undefined
*** Exception: Prelude.undefined
```

Задача 1.7. Докажете, че за всяко $a \in \mathbb{N}_\perp$, изображението $\texttt{const}_a^n \in [\mathbb{N}_\perp^n \xrightarrow{\text{н}} \mathbb{N}_\perp]$ е непрекъснато, където

$$\texttt{const}_a^n(x) \stackrel{\text{деф}}{=} a.$$

Задача 1.8. Докажете, че за всяко n и $i < n$, изображението $\texttt{proj}_i^{\mathcal{A},n} \in [\mathcal{A}^n \xrightarrow{\text{н}} \mathcal{A}]$, където

$$\texttt{proj}_i^{\mathcal{A},n}(a_0, \dots, a_{n-1}) \stackrel{\text{деф}}{=} a_i.$$

Ако от контекста е ясно кои са $\mathcal{A}$ и n, ще пишем просто $\mathtt{proj}_i$ вместо $\mathtt{proj}_i^{\mathcal{A},n}$.

Задача 1.9. Докажете, че $\mathtt{if}_{\mathcal{A}} \in [\mathbb{N}_\perp \times \mathcal{A} \times \mathcal{A} \stackrel{\text{н}}{\to} \mathcal{A}]$, където

$$\mathtt{if}_{\mathcal{A}}(b, a_1, a_2) \stackrel{\text{деф}}{=} \begin{cases} a_1, & \text{ако } b \in \mathbb{N}^+ \\ a_2, & \text{ако } b = 0 \\ \perp, & \text{ако } b = \perp. \end{cases}$$

Упътване. Докажете, че $\mathtt{if}_{\mathcal{A}}$ е непрекъснато изображение по всеки от аргументите си поотделно. □

Твърдение 1.8. Ако $f \in [\mathcal{A} \stackrel{\text{н}}{\to} B]$ и $g \in [\mathcal{B} \stackrel{\text{н}}{\to} \mathcal{C}]$, то $g \circ f \in [\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{C}]$, където

$$(g \circ f)(a) \stackrel{\text{деф}}{=} g(f(a)).$$

Упътване. Нека $(a_i)_{i=0}^\infty$ е верига в $\mathcal{A}$. Да обърнем внимание, че понеже $f \in [\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{B}]$, то f е монотонно изображение и тогава $(f(a_i))_{i=0}^\infty$ е верига в $\mathcal{B}$. Тогава:

$$\begin{aligned} (g \circ f)(\bigsqcup_i a_i) &= g(f(\bigsqcup_i a_i)) && \text{// от деф.} \\ &= g(\bigsqcup_i f(a_i)) && \text{// } f \text{ е непр.} \\ &= \bigsqcup_i g(f(a_i)) && \text{// } g \text{ е непр.} \end{aligned}$$

□

Задача 1.10. Докажете, че изображението

$$(.) : [\mathcal{B} \stackrel{\text{н}}{\to} \mathcal{C}] \times [\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{B}] \to [\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{C}]$$

е непрекъснато, където $(.)(f, g) = f \circ g$. С други думи, $(.)$ е елемент на областта на Скот

$$[[\mathcal{B} \stackrel{\text{н}}{\to} \mathcal{C}] \times [\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{B}] \stackrel{\text{н}}{\to} [\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{C}]].$$

Упътване. Използвайте *Твърдение* 1.7. □

Композицията на две функции е стандартна операция в хаскел.

```
ghci> :t (.)
(.) :: (b -> c) -> (a -> b) -> a -> c
```

Твърдение 1.9. Нека $f \in [\mathcal{A} \xrightarrow{\text{н}} \mathcal{B}]$ и $g \in [\mathcal{A} \xrightarrow{\text{н}} \mathcal{C}]$. Тогава $h \in [\mathcal{A} \xrightarrow{\text{н}} \mathcal{B} \times \mathcal{C}]$, където

$$h(a) \stackrel{\text{деф}}{=} \langle f(a), g(a) \rangle.$$

В такъв случай ще означаваме $h = f \times g$.

Доказателство. Нека $(a_i)_{i=0}^{\infty}$ е верига в $\mathcal{A}$. Тогава:

$$\begin{aligned}
h(\bigsqcup_i a_i) &= \langle f(\bigsqcup_i a_i), g(\bigsqcup_i a_i) \rangle && \text{// от деф.} \\
&= \langle \bigsqcup_i f(a_i), \bigsqcup_i g(a_i) \rangle && \text{// } f \text{ и } g \text{ са непр.} \\
&= \bigsqcup_i \langle f(a_i), g(a_i) \rangle && \text{// от \textit{Твърдение} 1.2} \\
&= \bigsqcup_i h(a_i) && \text{// от деф.}
\end{aligned}$$

□

Задача 1.11. Нека $f_i \in [\mathcal{A} \xrightarrow{\text{н}} \mathcal{B}_i]$, за $i = 1, \ldots, n$. Тогава докажете, че

✍ Докажете сами!

$$g \in [\mathcal{A} \xrightarrow{\text{н}} \prod_{i=1}^{n} \mathcal{B}_i],$$

където

$$g(a) \stackrel{\text{деф}}{=} \langle f_1(a), f_2(a), \ldots, f_n(a) \rangle.$$

В такъв случай ще означаваме $g = f_1 \times f_2 \cdots \times f_n$.

Ясно е, че може да се обобщи за произволно крайно декартово произведение.

Задача 1.12. Докажете, че изображението

$$\texttt{cross} : [\mathcal{A} \xrightarrow{\text{н}} \mathcal{B}] \times [\mathcal{A} \xrightarrow{\text{н}} \mathcal{C}] \to [\mathcal{A} \xrightarrow{\text{н}} \mathcal{B} \times \mathcal{C}]$$

е непрекъснато, където

$$\texttt{cross}(f, g) \stackrel{\text{деф}}{=} f \times g.$$

Можем да използваме операцията за декартово произведение на хаскел по следния начин.

```
ghci> import Control.Arrow
ghci> :t (&&&)
(&&&) :: Arrow a => a b c -> a b c' -> a b (c, c')
ghci> (&&&) (\x->x+1) (\x->x+2) 2
(3,4)
ghci> (&&&) (\x->x^2) (\x->x^3) 2
(4,8)
```

Определение 1.3. Нека $\mathcal{A}$ и $\mathcal{B}$ са области на Скот. Дефинираме изображението

$$\mathtt{eval} : [\mathcal{A} \xrightarrow{\text{н}} \mathcal{B}] \times \mathcal{A} \to \mathcal{B},$$

по следния начин:

$$\mathtt{eval}(f, a) \stackrel{\text{деф}}{=} f(a).$$

[4, стр. 186]

Задача 1.13. Докажете, че `eval` е непрекъснато изображение, т.е.

$$\mathtt{eval} \in [[\mathcal{A} \xrightarrow{\text{н}} \mathcal{B}] \times \mathcal{A} \xrightarrow{\text{н}} \mathcal{B}].$$

Доказателство. Според *Твърдение* 1.7, достатъчно е да докажем, че `eval` е непрекъснато изображение по всеки от двата си аргумента поотделно.

Операцията `eval` често се означава и като `apply`.

Първо, нека $(f_n)_{n=0}^{\infty}$ е верига от елементи на $[\mathcal{A} \xrightarrow{\text{н}} \mathcal{B}]$ и d е произволен елемент на $\mathcal{A}$. Тогава

$$\mathtt{eval}(\bigsqcup_n f_n, a) = (\bigsqcup_n f_n)(a) = \bigsqcup_n \{f_n(a)\} = \bigsqcup_n \mathtt{eval}(f_n, a),$$

т.е. изображението `eval` е непрекъснато по първия си аргумент.

Нека сега $(a_n)_{n=0}^{\infty}$ е верига от елементи на $\mathcal{D}$. Тогава за произволен елемент f на $[\mathcal{A} \xrightarrow{\text{н}} \mathcal{B}]$ получаваме, че

$$\mathtt{eval}(f, \bigsqcup_n a_n) = f(\bigsqcup_n a_n) = \bigsqcup_n \{f(a_n)\} = \bigsqcup_n \mathtt{eval}(f, a_n).$$

□

Операцията `eval` е стандартна операция в `хаскел`.

```
ghci> :t ($)
($) :: (a -> b) -> a -> b
ghci> ($) (+1) 3
4
ghci> ($) head [1..10]
1
```

Определение 1.4. Нека $\mathcal{A}$, $\mathcal{B}$ и $\mathcal{C}$ са области на Скот. Изображението

$$\mathtt{curry} : [\mathcal{A} \times \mathcal{B} \to \mathcal{C}] \to [\mathcal{A} \to [\mathcal{B} \to \mathcal{C}]],$$

е дефинирано като

$$\mathtt{curry}(f)(a)(b) \stackrel{\text{деф}}{=} f(a, b).$$

[4, стр. 187]

Твърдение 1.10. Ако f е непрекъснато изображение, то $\mathtt{curry}(f)$ е непрекъснато изображение, т.е. $\mathtt{curry}(f) \in [\mathcal{A} \xrightarrow{\text{н}} [\mathcal{B} \xrightarrow{\text{н}} \mathcal{C}]]$. Освен това,

$$\mathtt{curry} \in [[\mathcal{A} \times \mathcal{B} \xrightarrow{\text{н}} \mathcal{C}] \xrightarrow{\text{н}} [\mathcal{A} \xrightarrow{\text{н}} [\mathcal{B} \xrightarrow{\text{н}} \mathcal{C}]]].$$

Доказателство. Първо да фиксираме $f \in [\mathcal{A} \times \mathcal{B} \overset{\text{н}}{\to} \mathcal{C}]$ и $a \in \mathcal{A}$. Ще докажем, че $\texttt{curry}(f)(a) \in [\mathcal{B} \overset{\text{н}}{\to} \mathcal{C}]$. Нека фиксираме верига $(b_i)_{i=0}^{\infty}$ от елементи на $\mathcal{B}$. Тогава

Ако означим $h \overset{\text{деф}}{=} \texttt{curry}(f)(a)$, то трябва да докажем, че за произволна верига $(b_i)_{i=0}^{\infty}$ в $\mathcal{B}$, $h(\bigsqcup_i b_i) = \bigsqcup_i \{h(b_i)\}$.

$$
\begin{aligned}
\texttt{curry}(f)(a)(\bigsqcup_i b_i) &\overset{\text{деф}}{=} f(a, \bigsqcup_i b_i) \\
&= \bigsqcup_i \{f(a, b_i)\} && \text{// } f \text{ е непр. по втория си аргумент} \\
&= \bigsqcup_i \{\texttt{curry}(f)(a)(b_i)\}. && \text{// опр. на } \texttt{curry}
\end{aligned}
$$

Второ, сега пък за фиксирано $f \in [\mathcal{A} \times \mathcal{B} \overset{\text{н}}{\to} \mathcal{C}]$ трябва да докажем, че $\texttt{curry}(f) \in [\mathcal{A} \overset{\text{н}}{\to} [\mathcal{B} \overset{\text{н}}{\to} \mathcal{C}]]$. За целта да разгледаме произволна верига $(a_i)_{i=0}^{\infty}$ от елементи на $\mathcal{A}$. Трябва да докажем, че

$$\texttt{curry}(f)(\bigsqcup_i a_i) = (\bigsqcup_i \texttt{curry}(f))(a_i).$$

Също така, да напомним, че за произволен елемент $b \in \mathcal{B}$,

Обърнете внимание, че $\texttt{curry}(f)(a_i)$ образуват верига в $[\mathcal{B} \overset{\text{н}}{\to} \mathcal{C}]$, откъдето следва, че $\bigsqcup_i \{\texttt{curry}(f)(a_i)\}$ е добре дефиниран елемент.

$$(\bigsqcup_i \{\texttt{curry}(f)(a_i)\})(b) \overset{\text{деф}}{=} \bigsqcup_i \{\texttt{curry}(f)(a_i)(b)\}. \tag{1.2}$$

Тогава

$$
\begin{aligned}
\texttt{curry}(f)(\bigsqcup_i a_i)(b) &= f(\bigsqcup_i a_i, b) && \text{// опр. на } \texttt{curry} \\
&= \bigsqcup_i f(a_i, b) && \text{// } f \text{ е непр. по първия аргумент} \\
&= \bigsqcup_i \{\texttt{curry}(f)(a_i)(b)\} && \text{// опр. на } \texttt{curry} \\
&= (\bigsqcup_i \{\texttt{curry}(f)(a_i)\})(b). && \text{// от (1.2)}
\end{aligned}
$$

Заключаваме, че $\texttt{curry}(f)(\bigsqcup_i a_i) = (\bigsqcup_i \texttt{curry}(f))(a_i)$.

Трето, остава да видим защо за произволна верига $(f_i)_{i=0}^{\infty}$ от елементи на $[\mathcal{A} \times \mathcal{B} \overset{\text{н}}{\to} \mathcal{C}]$ е изпълнено, че

$$\texttt{curry}(\bigsqcup_i f_i) = \bigsqcup_i \{\texttt{curry}(f_i)\}.$$

За произволен елемент $a \in \mathcal{A}$ имаме, че

$$(\bigsqcup_i \texttt{curry}(f_i))(a) = \bigsqcup_i \{\texttt{curry}(f_i)(a)\}.$$

За произволен елемент $b \in \mathcal{B}$ имаме, че

Ако $h_i \overset{\text{деф}}{=} \texttt{curry}(f_i)(a)$, то е ясно, че $(\bigsqcup_i h_i)(b) = \bigsqcup_i \{h_i(b)\}$.

$$(\bigsqcup_i \texttt{curry}(f_i)(a))(b) = \bigsqcup_i \{\texttt{curry}(f_i)(a)(b)\}.$$

Комбинирайки предишните две равенства, получаваме, че за произволни $a \in \mathcal{A}$ и $b \in \mathcal{B}$,

$$
\begin{aligned}
(\bigsqcup_i \mathtt{curry}(f_i))(a)(b) &= \bigsqcup_i \{\mathtt{curry}(f_i)(a)(b)\} \\
&= \bigsqcup_i \{f_i(a,b)\} && \text{// от опр. на } \mathtt{curry} \\
&= (\bigsqcup_i f_i)(a,b) \\
&= \mathtt{curry}(\bigsqcup_i f_i)(a)(b). && \text{// от опр. на } \mathtt{curry}
\end{aligned}
$$

□

Определение 1.5. Нека $\mathcal{A}$, $\mathcal{B}$ и $\mathcal{C}$ са области на Скот. Изображението

$$\mathtt{uncurry} : [[\mathcal{A} \to [\mathcal{B} \to \mathcal{C}]] \to [\mathcal{A} \times \mathcal{B} \to \mathcal{C}]],$$

е дефинирано като

$$\mathtt{uncurry}(f)(a,b) \stackrel{\text{деф}}{=} f(a)(b).$$

Задача 1.14. Докажете, че ако f е непрекъснато изображение, то $\mathtt{uncurry}(f)$ е непрекъснато изображение, т.е. $\mathtt{uncurry}(f) \in [\mathcal{A} \times \mathcal{B} \xrightarrow{\text{н}} \mathcal{C}]$. Освен това, докажете, че

$$\mathtt{uncurry} \in [[\mathcal{A} \xrightarrow{\text{н}} [\mathcal{B} \xrightarrow{\text{н}} \mathcal{C}]] \xrightarrow{\text{н}} [\mathcal{A} \times \mathcal{B} \xrightarrow{\text{н}} \mathcal{C}]].$$

Всъщност, хаскел има функциите curry и uncurry вградени в стандартната библиотека:

```
ghci> :t curry
curry :: ((a, b) -> c) -> a -> b -> c
ghci> :t uncurry
uncurry :: (a -> b -> c) -> (a, b) -> c
```

[4, стр. 177].

Задача 1.15. Докажете, че изображенията $\mathtt{fst} : \mathcal{A} \times \mathcal{B} \to \mathcal{A}$ и $\mathtt{snd} : \mathcal{A} \times \mathcal{B} \to \mathcal{B}$ са непрекъснати, където $\mathtt{fst}(a,b) \stackrel{\text{деф}}{=} a$ и $\mathtt{snd}(a,b) \stackrel{\text{деф}}{=} b$.

```
ghci> :t fst
fst :: (a, b) -> a
ghci> :t snd
snd :: (a, b) -> b
```

1.6 Най-малки неподвижни точки

- Да фиксираме произволна област на Скот $\mathcal{A} = (A, \sqsubseteq, \perp)$ и да разгледаме едно изображение $f : \mathcal{A} \to \mathcal{A}$. Казваме, че $a \in \mathcal{A}$ е **неподвижна точка** на f, ако $f(a) = a$.
- Казваме, че $a \in \mathcal{A}$ е **най-малката неподвижна точка** на f, ако:
  - a е неподвижна точка, т.е. $f(a) = a$;
  - за всяко $b \in \mathcal{A}$ със свойството, че $f(b) = b$ имаме $a \sqsubseteq b$.
  - Ще означаваме най-малката неподвижна точка на f като $\texttt{lfp}(f)$.

На англ. *least fixed point*

Теорема 1.5 (Клини). Нека $\mathcal{A}$ е област на Скот. Всяко изображение $f \in [\mathcal{A} \xrightarrow{\text{н}} \mathcal{A}]$ притежава най-малка неподвижна точка.

Доказателство. Определяме монотонно растяща редица от елементи на $\mathcal{A}$ по следния начин:

В [14] се нарича теорема на Кнастер-Тарски. Според уикипедия е теорема на Клини. Теоремата на Кнастер-Тарски е по-силна и говори за монотонни изображения в решетки.

$$
\begin{aligned}
a_0 &\stackrel{\text{деф}}{=} \perp && // = f^0(\perp)\\
a_{n+1} &\stackrel{\text{деф}}{=} f(a_n) && // = f^{n+1}(\perp).
\end{aligned}
$$

Първо ще докажем с индукция по n, че $(a_n)_{n=0}^{\infty}$ е верига. Ясно е, че $a_0 \sqsubseteq a_1$. Да приемем, че $a_n \sqsubseteq a_{n+1}$. Тогава, понеже всяко непрекъснато изображение е монотонно, то имаме, че

$$\underbrace{f(a_n)}_{a_{n+1}} \sqsubseteq \underbrace{f(a_{n+1})}_{a_{n+2}}.$$

Нека $a \stackrel{\text{деф}}{=} \bigsqcup_i a_i$. Тогава

$$
\begin{aligned}
f(a) &= f(\bigsqcup_i a_i) && // \; a \stackrel{\text{деф}}{=} \bigsqcup_i a_i\\
&= \bigsqcup_i f(a_i) && // \; f \text{ е непрекъсната}\\
&= \bigsqcup_i a_{i+1} && // \; a_{i+1} = f(a_i)\\
&= \bigsqcup_i a_i && // \text{ защото } (a_i)_{i=0}^{\infty} \text{ е верига}\\
&= a.
\end{aligned}
$$

Така доказахме, че a е *неподвижна точка* на f. Остана да видим, че е най-малката неподвижна точка на f.

Нека $b = f(b)$. С индукция по n ще докажем, свойството $(\forall n)[a_n \sqsubseteq b]$.

- За $n = 0$ е очевидно.
- Да приемем, че $a_n \sqsubseteq b$. Тогава $a_{n+1} \stackrel{\text{деф}}{=} f(a_n) \sqsubseteq f(b) = b$, защото f е монотонно изображение.

Така доказахме, че b е горна граница на веригата $(a_n)_{n=0}^{\infty}$. Заключаваме, че $a \stackrel{\text{деф}}{=} \bigsqcup_n a_n \sqsubseteq b$. Следователно, a е *най-малката неподвижна точка* на f, т.е. $a = \mathtt{lfp}(f)$. □

Задача 1.16. Покажете, че съществува област на Скот $\mathcal{A}$ и изображение $f \in [\mathcal{A} \xrightarrow{\text{м}} \mathcal{A}]$, което притежава най-малка неподвижна точка, но тя не е $\bigsqcup_n f^n(\perp^{\mathcal{A}})$.

Упътване. Вземете областта на Скот $\mathcal{A}$ и монотонното изображение f както в *Твърдение* 1.5. Лесно се вижда, че f не е непрекъснато изображение, защото $\bigsqcup_n a_n = a_\omega$, и тогава:

$$f(\bigsqcup_n a_n) = f(a_\omega) = b \neq a_\omega = \bigsqcup_n a_{n+1} = \bigsqcup_n \{f(a_n)\}.$$

Според дефиницията на изображението f, единствената неподвижна точка на f е елементът b. Това означава, че b е също и най-малката неподвижна точка. Това е пример за монотонно изображение, което не е непрекъснато, но притежава най-малка неподвижна точка b, макар и тя да не е $a_\omega = \bigsqcup_n f^n(a_0)$. □

Имаме, че $f^n(a_0) = a_n$.

Да се обясни къде се използва това твърдение.

Определение 1.6. Нека $\mathcal{A}$ е област на Скот и $f \in [\mathcal{A} \to \mathcal{A}]$. Ще казваме, че е елементът $a \in \mathcal{A}$ е **преднеподвижна точка** на f, ако $f(a) \sqsubseteq a$. Да положим

$$\mathtt{Pref}(f) \stackrel{\text{деф}}{=} \{a \in \mathcal{A} \mid f(a) \sqsubseteq a\}.$$

Твърдение 1.11. За всяко $f \in [\mathcal{A} \xrightarrow{\text{н}} \mathcal{A}]$ е изпълнено, че

$$(\forall a \in \mathtt{Pref}(f))[\mathtt{lfp}(f) \sqsubseteq a].$$

Това означава, че $\mathtt{lfp}(f)$ е най-малката преднеподвижна точка на f.

Доказателство. Знаем от Теоремата на Клини, че $\mathtt{lfp}(f) = \bigsqcup_n f^n(\perp)$. Също така знаем, че $(b_n)_{n=0}^{\infty}$ е верига, където за улеснение сме положили $b_n \stackrel{\text{деф}}{=} f^n(\perp)$. Ясно е също, че $\mathtt{Pref}(f) \neq \emptyset$, защото $\mathtt{lfp}(f) \in \mathtt{Pref}(f)$. Да фиксираме прозиволен елемент $a \in \mathtt{Pref}(f)$. С индукция по n ще докажем, че $b_n \sqsubseteq a$ за всяко n.

- За $n = 0$ е очевидно, защото тогава $b_0 \stackrel{\text{деф}}{=} \bot \sqsubseteq a$.
- Да приемем, че $b_n \sqsubseteq a$. Ще докажем, че $b_{n+1} \sqsubseteq a$. Но това е лесно.

$$\begin{aligned} b_{n+1} &= f(b_n) && \text{// от деф. на } b_{n+1} \\ &\sqsubseteq f(a) && \text{// } b_n \sqsubseteq a \ \&\ f \text{ е мон.} \\ &\sqsubseteq a && \text{// } a \in \texttt{Pref}(f). \end{aligned}$$

Така доказахме, че за всяко n, $b_n \sqsubseteq a$, откъдето следва, че a е горна граница за веригата $(b_n)_{n=0}^{\infty}$, откъдето директно получаваме, че

$$\texttt{lfp}(f) = \bigsqcup_n b_n \sqsubseteq a.$$

□

Задача 1.17. Нека $\mathcal{A}$ е област на Скот и нека $f \in [\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{A}]$. Да разгледаме множеството $B = \{a \in \mathcal{A} \mid f(a) = a\}$. Вярно ли е, че $\mathcal{B} = (B, \sqsubseteq, \texttt{lfp}(f))$ е област на Скот. Обосновете се!

Задача 1.18. Нека $f \in [\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{A}]$. Да разгледаме множеството

$$B = \{a \in \mathcal{A} \mid f(a) \sqsubseteq a\}.$$

Вярно ли е, че

$$\mathcal{B} = (B, \sqsubseteq^{\mathcal{A}}, \texttt{lfp}(f))$$

е област на Скот? Обосновете се!

Задача 1.19. Да разгледаме множеството

$$B = \{f \in [\mathcal{A} \stackrel{\text{м}}{\to} \mathcal{A}] \mid f \circ f = f\}.$$

Вярно ли е, че

$$\mathcal{B} = (B,\ \sqsubseteq,\ \lambda x.\bot^{\mathcal{A}})$$

е област на Скот, където

$$f \sqsubseteq g \stackrel{\text{деф}}{=} (\forall a \in \mathcal{A})[f(a) \sqsubseteq^{\mathcal{A}} g(a)]?$$

Обосновете се!

Задача 1.20. Да разгледаме произволна област на Скот $\mathcal{A}$ и изображение $f \in [\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{A}]$. Докажете, че:

$$\texttt{lfp}(f) = \texttt{lfp}(f \circ f).$$

[13, стр. 131] и задача в Кеймбридж 2020 г. [5]

Задача 1.21. Нека $f \in [\mathcal{A} \overset{\text{н}}{\to} \mathcal{B}]$ и $g \in [\mathcal{B} \overset{\text{н}}{\to} \mathcal{A}]$. Докажете, че

- $\mathtt{lfp}(g \circ f) \sqsubseteq g(\mathtt{lfp}(f \circ g))$;
- $f(\mathtt{lfp}(g \circ f)) \sqsubseteq \mathtt{lfp}(f \circ g)$.

Оттук заключете, че

$$\mathtt{lfp}(g \circ f) = g(\mathtt{lfp}(f \circ g)) \text{ и } f(\mathtt{lfp}(g \circ f)) = \mathtt{lfp}(f \circ g).$$

Изображения h, за които $h(\perp^{\mathcal{A}}) = \perp^{\mathcal{B}}$ се наричат точни. На англ. strict.

Задача 1.22. Нека $\mathcal{A}$ и $\mathcal{B}$ са области на Скот и нека $f \in [\mathcal{A} \overset{\text{н}}{\to} \mathcal{A}]$, $g \in [\mathcal{B} \overset{\text{н}}{\to} \mathcal{B}]$, и $h \in [\mathcal{A} \overset{\text{н}}{\to} \mathcal{B}]$, за които е изпълнено свойството, че $h \circ f = g \circ h \in [\mathcal{A} \overset{\text{н}}{\to} \mathcal{B}]$. Докажете, че ако h е такава, че $h(\perp^{\mathcal{A}}) = \perp^{\mathcal{B}}$, то е изпълнено, че:

$$\mathtt{lfp}(g) = h(\mathtt{lfp}(f)).$$

Много от задачите са от [1, стр. 31]

Задача 1.23. Да разгледаме произволна област на Скот $\mathcal{A}$ и непрекъснатите изображения

$$\Gamma, \Delta \in [\mathcal{A} \overset{\text{н}}{\to} \mathcal{A}].$$

Винаги ли е вярно, че

$$\mathtt{lfp}(\Gamma \circ \Delta) \sqsubseteq \mathtt{lfp}(\Gamma) \circ \mathtt{lfp}(\Delta)?$$

Обосновете се!

Упътване. Не. Ще дадем контрапример. Нека $\mathcal{A} = [\mathbb{N}_\perp \overset{\text{н}}{\to} \mathbb{N}_\perp]$. Нека например да разгледаме

$$\Delta(f)(x) \overset{\text{деф}}{=} f(x+1)$$
$$\Gamma(f)(x) \overset{\text{деф}}{=} \begin{cases} 0, & x \neq \perp \\ \perp, & x = \perp. \end{cases}$$

Съобразете сами, че Δ и Γ са непрекъснати. Да положим $f_\Gamma \overset{\text{деф}}{=} \mathtt{lfp}(\Gamma)$ и $f_\Delta \overset{\text{деф}}{=} \mathtt{lfp}(\Delta)$. Ясно е, че

$$f_\Delta(x) = \perp$$
$$f_\Gamma(x) = \begin{cases} 0, & x \neq \perp \\ \perp, & x = \perp. \end{cases}$$

Тогава за произволно $x \in \mathbb{N}_\perp$,

$$(f_\Gamma \circ f_\Delta)(x) = f_\Gamma(f_\Delta(x)) = f_\Gamma(\perp) = \perp.$$

От друга страна, понеже $(\Gamma \circ \Delta)(f) = \Gamma(\Delta(f))$, то

$$(\Gamma \circ \Delta)(f)(x) = \Gamma(\Delta(f))(x) = \begin{cases} 0, & x \neq \perp \\ \perp, & x = \perp. \end{cases}$$

Лесно се съобразява, че

$$\texttt{lfp}(\Gamma \circ \Delta)(x) = \begin{cases} 0, & x \neq \perp \\ \perp, & x = \perp. \end{cases}$$

Заключаваме, че

$$\texttt{lfp}(\Gamma) \circ \texttt{lfp}(\Delta) \sqsubseteq \texttt{lfp}(\Gamma \circ \Delta),$$

но

$$\texttt{lfp}(\Gamma \circ \Delta) \not\sqsubseteq \texttt{lfp}(\Gamma) \circ \texttt{lfp}(\Delta).$$

□

Задача 1.24. Да разгледаме произволна област на Скот $\mathcal{A}$ и непрекъснатите изображения

$$\Gamma, \Delta \in [\mathcal{A} \overset{\text{н}}{\to} \mathcal{A}].$$

Винаги ли е вярно, че:

$$\texttt{lfp}(\Gamma) \circ \texttt{lfp}(\Delta) \sqsubseteq \texttt{lfp}(\Gamma \circ \Delta)?$$

Обосновете се!

Упътване. Ще дадем контрапример. Нека $\mathcal{A} = [\mathbb{N}_\perp \overset{\text{н}}{\to} \mathbb{N}_\perp]$. Нека например да разгледаме

$$\Delta(f)(x) \overset{\text{деф}}{=} 0$$

$$\Gamma(f)(x) \overset{\text{деф}}{=} \begin{cases} 0, & x = 0 \\ \perp, & \text{иначе}. \end{cases}$$

Съобразете сами, че Δ и Γ са непрекъснати. Да положим $f_\Gamma \overset{\text{деф}}{=} \texttt{lfp}(\Gamma)$ и $f_\Delta \overset{\text{деф}}{=} \texttt{lfp}(\Delta)$. Ясно е, че

$$f_\Delta(x) = 0$$

$$f_\Gamma(x) = \begin{cases} 0, & x = 0 \\ \perp, & \text{иначе}. \end{cases}$$

Тогава за произволен елемент $x \in \mathbb{N}_\perp$,

$$(f_\Gamma \circ f_\Delta)(x) = f_\Gamma(f_\Delta(x)) = f_\Gamma(0) = 0.$$

От друга страна, понеже $(\Gamma \circ \Delta)(f) = \Gamma(\Delta(f))$, то

$$(\Gamma \circ \Delta)(f)(x) = \Gamma(\Delta(f))(x) = \begin{cases} 0, & x = 0 \\ \perp, & \text{иначе}. \end{cases}$$

Лесно се съобразява, че

$$\texttt{lfp}(\Gamma \circ \Delta)(x) = \begin{cases} 0, & x = 0 \\ \perp, & \text{иначе}. \end{cases}$$

Заключаваме, че

$$\texttt{lfp}(\Gamma \circ \Delta) \sqsubseteq \texttt{lfp}(\Gamma) \circ \texttt{lfp}(\Delta),$$

но

$$\texttt{lfp}(\Gamma) \circ \texttt{lfp}(\Delta) \not\sqsubseteq \texttt{lfp}(\Gamma \circ \Delta).$$

$\square$

1.7 Оператор за най-малка неподвижна точка

Задача 1.25. Нека $(f_i)_{i=0}^{\infty}$ е верига от елементи на $[\mathcal{A} \stackrel{\text{м}}{\to} \mathcal{A}]$. Докажете, че за произволно фиксирано n, $(f_i^n)_{i=0}^{\infty}$ е верига.

Да напомним, че $f^0(a) = a$ и $f^{n+1}(a) = f(f^n(a))$.

Упътване. Първо докажете, че за произволни $g, h \in [\mathcal{A} \to \mathcal{A}]$ и $f \in [\mathcal{A} \stackrel{\text{м}}{\to} \mathcal{A}]$ е изпълнено, че ако $g \sqsubseteq h$, то $f \circ g \sqsubseteq f \circ h$.

След това може да престъпите към индукция по n. □

Забележка 1.1. Обърнете внимание, че лесно може да се намери $f \in [\mathcal{A} \stackrel{\text{м}}{\to} \mathcal{A}]$, за която $f \not\sqsubseteq f \circ f$. Например, нека разгледаме $f \in [\mathbb{N}_\perp \stackrel{\text{м}}{\to} \mathbb{N}_\perp]$, където

$$f(x) \stackrel{\text{деф}}{=} \begin{cases} 6, & \text{ако } x = 5 \\ \perp, & \text{иначе.} \end{cases}$$

Лесно се съобразява, че за всяко $x \in \mathbb{N}_\perp$, $f(f(x)) = \perp$. Оттук веднага следва, че $f \not\sqsubseteq f \circ f$.

Сега можем да направим и заключението, че не за всяка $f \in [\mathcal{A} \stackrel{\text{м}}{\to} \mathcal{A}]$ е изпълнено, че $(f^n)_n^{\infty}$ е верига. От друга страна, от доказателството на Теоремата на Клини знаем, че за всяка $f \in [\mathcal{A} \stackrel{\text{м}}{\to} \mathcal{A}]$ е изпълнено, че $(f^n(\perp))_n^{\infty}$ е верига.

Теорема 1.6. Нека $\mathcal{A}$ е област на Скот и нека $f \in [\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{A}]$. Тогава изображението $Y : [\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{A}] \to \mathcal{A}$, определено като

$$Y(f) = \mathtt{lfp}(f),$$

е непрекъснато, т.е. $Y \in [[\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{A}] \stackrel{\text{н}}{\to} \mathcal{A}]$.

Задача в Кеймбридж 2022 г. [5]. Доказателството в [4, стр. 188] е малко по-различно. Знаем от *Теорема* 1.5, че най-малката неподвижна точка на f е елемента $\bigsqcup_n f^n(\perp^{\mathcal{A}})$, т.е. $\mathtt{lfp}(f) = \bigsqcup_n f^n(\perp^{\mathcal{A}})$.

Доказателство. Нека да вземем една верига $(f_n)_{n=0}^{\infty}$ от непрекъснати изображения. Нашата цел е да докажем, че

$$Y(\bigsqcup_n f_n) = \bigsqcup_n Y(f_n).$$

Да означим с h точната горна граница на $(f_n)_{n=0}^{\infty}$. Знаем, че $h(a) = \bigsqcup_n f_n(a)$, а от *Теорема* 1.4 знаем, че h е непрекъснато изображение.

Записът ще стане много тромав, ако вместо h използваме означението $\bigsqcup_n f_n$.

Твърдение 1.12. За всяко $k \geq 0$ е изпълнено, че $(\bigsqcup f_i)^k = \bigsqcup_i f_i^k$.

Доказателство. Ще докажем твърдението с индукция по k, като случая $k = 0$ е ясен, защото $(\bigsqcup_i f_i)^0 = id = \bigsqcup_i f_i^0 = \bigsqcup_i id$. Нека приемем, че твърдението е вярно за произволно k. Ще докажем, че твърдението е вярно за $k + 1$.

$$(\bigsqcup_i f_i)^{k+1} = (\,.\,)((\bigsqcup_i f_i)^k, \bigsqcup_i f_i)$$

$$
\begin{aligned}
&= (\,.\,)(\bigsqcup_i f_i^k, \bigsqcup_i f_i) && \text{// от } \textbf{(И.П.)} \\
&= \bigsqcup_i (\,.\,)(f_i^k, f_i) && \text{// } (\,.\,) \text{ е непр.} \\
&= \bigsqcup_i f_i^{k+1}
\end{aligned}
$$

С това твърдението е доказано. □

Сега вече сме готови да докажем непрекъснатостта на Y. Имаме, че:

$$
\begin{aligned}
Y(\bigsqcup_i f_i) &= \bigsqcup_m (\bigsqcup_i f_i)^m(\perp^{\mathcal{A}}) && \text{// от опр. на } Y \\
&= \bigsqcup_m (\bigsqcup_i f_i^m(\perp^{\mathcal{A}})) && \text{// от горното твърдение}
\end{aligned}
$$

Да положим $e_{m,n} = f_n^m(\perp^{\mathcal{A}})$. Отново лесно се съобразява, че

$$
m \le m' \;\&\; n \le n' \;\Rightarrow\; e_{m,n} \sqsubseteq^{\mathcal{A}} e_{m',n'}.
$$

Получаваме, че

$$
\begin{aligned}
Y(\bigsqcup_n f_n) &= \bigsqcup_m (\bigsqcup_n f_n^m(\perp^{\mathcal{A}})) && \text{// от горното равенство} \\
&= \bigsqcup_m (\bigsqcup_n e_{m,n}) && \text{// от опр. на } e_{m,n} \\
&= \bigsqcup_n (\bigsqcup_m e_{m,n}) && \text{// от } \textit{Лема } 1.1 \\
&= \bigsqcup_n (\bigsqcup_m f_n^m(\perp^{\mathcal{A}})) = \bigsqcup_n Y(f_n). && \text{// от опр. на } Y.
\end{aligned}
$$

□

Нека да видим как можем да дефинираме на хаскел операторът Y и как можем да го използваме.

Добре е да погледнете това.

```
ghci> fact x = if x == 0 then 1 else x * fact(x-1)
ghci> fact 5
120
ghci> fix f = x where x = f x
ghci> :t f
fix :: (t -> t) -> t
ghci> fact' = fix \f x -> if x == 0 then 1 else x * f(x-1)
ghci> fact' 5
120
ghci> gamma f = \x -> if x == 0 then 1 else x * f(x-1)
ghci> :t gamma
```

```
(t -> t) -> t -> t
ghci> fact'' = fix gamma
ghci> fact'' 5
120
ghci> fix' f = x where x = f(f(f(x)))
ghci> fact''' = fix' gamma
ghci> fact''' 5
120
```

Задача 1.26. Нека $f \in [\mathcal{A} \times \mathcal{B} \stackrel{\text{н}}{\to} \mathcal{A}]$. Да положим $f_b(x) \stackrel{\text{деф}}{=} f(x, b)$, за произволно $b \in \mathcal{B}$. Ясно е, че $f_b \in [\mathcal{A} \stackrel{\text{н}}{\to} \mathcal{A}]$. Да положим $\boldsymbol{a}(b) = \mathtt{lfp}(f_b)$. Докажете, че $\boldsymbol{a} \in [\mathcal{B} \stackrel{\text{н}}{\to} \mathcal{A}]$.

1.8 Най-малко решение на система от уравнения

Нека $\mathcal{A}_1, \dots, \mathcal{A}_n$ са области на Скот и да разгледаме изображенията

$$f_i : \prod_{k=1}^{n} \mathcal{A}_k \to \mathcal{A}_i,$$

за $i = 1, \dots, n$. Казваме, че $\bar{a} = \langle a_1, \dots, a_n \rangle$ е **решение на системата**

$$\bigstar = \begin{cases} x_1 = f_1(x_1, \dots, x_n) \\ \vdots \\ x_n = f_n(x_1, \dots, x_n), \end{cases}$$

ако са в сила равенствата:

$$\left| \begin{array}{lcl} a_1 & = & f_1(a_1, \dots, a_n) \\ & \vdots & \\ a_n & = & f_n(a_1, \dots, a_n). \end{array} \right.$$

Казваме, че $\bar{a}$ е **най-малкото решение** на системата $\bigstar$, ако за всяко друго решение $\bar{b}$ е изпълнено, че $\bar{a} \sqsubseteq \bar{b}$.

Теорема 1.7. За произволни изображения $f_i \in [\prod_{k=1}^{n} \mathcal{A}_k \xrightarrow{\text{н}} \mathcal{A}_i]$, за $i = 1, \dots, n$, системата

$$\left| \begin{array}{lcl} x_1 & = & f_1(x_1, \dots, x_n) \\ & \vdots & \\ x_n & = & f_n(x_1, \dots, x_n), \end{array} \right.$$

притежава най-малко решение.

Доказателство. Първо да дефинираме изображението

$$g \stackrel{\text{деф}}{=} f_1 \times \dots \times f_n : \prod_{k=1}^{n} \mathcal{A}_k \to \prod_{k=1}^{n} \mathcal{A}_k,$$

като

$$g(\overline{a}) \stackrel{\text{деф}}{=} \langle f_1(\overline{a}), \dots, f_n(\overline{a}) \rangle,$$

което е непрекъснато според *Задача* 1.11. От Теоремата на Клини знаем, че g притежава най-малка неподвижна точка $\bar{a} = \langle a_1, \dots, a_n \rangle$. Ще проверим, че $\bar{a}$ е най-малкото решение на системата.

- Понеже $\overline{a}$ е неподвижна точка на g, то

$$\begin{aligned} g(a_1, \dots, a_n) &= \langle f_1(\overline{a}), \dots, f_n(\overline{a}) \rangle && \text{// от деф. на } g \\ &= \langle a_1, \dots, a_n \rangle && \text{// } \overline{a} \text{ е неподвижна точка.} \end{aligned}$$

  Оттук директно следва, че $f_i(\bar{a}) = a_i$, за $i = 1, \dots, n$, и следователно $\overline{a}$ е решение на системата.

- Нека $\overline{b} = \langle b_1, \dots, b_n \rangle$ е друго решение на системата, т.е. $f_i(\overline{b}) = b_i$, за $i = 1, \dots, n$. Тогава $g(\overline{b}) = \langle f_1(\overline{b}), \dots, f_n(\overline{b}) \rangle = \overline{b}$. Следователно $\overline{b}$ е неподвижна точна на g. Понеже $\overline{a} = \mathtt{lfp}(g)$, то $\overline{a} \sqsubseteq \overline{b}$.

Така достигнахме до извода, че $\overline{a}$ е най-малкото решение на системата. □

Забележка 1.2. Да разгледаме изображенията $f \in [\mathcal{A} \times \mathcal{B} \overset{\text{н}}{\to} \mathcal{A}]$, $g \in [\mathcal{B} \overset{\text{н}}{\to} \mathcal{B}]$ и системата от две уравнения:

$$\left| \begin{array}{lcl} f(x,y) & = & x \\ g(y) & = & y. \end{array} \right.$$

За да можем директно да се позовем на *Теорема* 1.7 и да твърдим, че тази система има най-малко решение, ние трябва да разгледаме следната модификация на системата:

$$\left| \begin{array}{lcl} f(x,y) & = & x \\ \hat{g}(x,y) & = & y, \end{array} \right.$$

където $\hat{g}(x,y) = g(y)$, т.е. добавяме един фиктивен аргумент, защото искаме всички изображения да имат равен брой аргументи.

Ще завършим този раздел с две твърдения, които ще улеснят нашите разсъждения при решаването на задачи.

Твърдение 1.13. Да разгледаме две изображения

$$\begin{aligned} & f \in [\mathcal{A} \times \mathcal{B} \overset{\text{н}}{\to} \mathcal{A}] \\ & g \in [\mathcal{B} \overset{\text{н}}{\to} \mathcal{B}], \end{aligned}$$

за които имаме системата от уравнения

$$\bigstar = \begin{cases} f(x,y) = x \\ g(y) = y. \end{cases}$$

Нека $b_0 = \mathtt{lfp}(g)$ и $a_0 = \mathtt{lfp}(\hat{f})$, където $\hat{f}(a) \overset{\text{деф}}{=} f(a, b_0)$. Тогава $\langle a_0, b_0 \rangle$ е най-малкото решение на системата $\bigstar$.

Доказателство.

- Първо, понеже $b_0 = \texttt{lfp}(g)$, то очевидно $g(b_0) = b_0$. Освен това, $a_0 = \texttt{lfp}(\hat{f})$, откъдето следва, че $a_0 = f(a_0, b_0)$. Ясно е, че $\langle a_0, b_0\rangle$ е решение на системата ★.
- Сега нека $\langle a, b\rangle$ е произволно решение на системата ★. Да видим, че $\langle a_0, b_0\rangle \sqsubseteq \langle a, b\rangle$.
  - Първо, ясно е, че $b = g(b)$. Понеже $b_0 = \texttt{lfp}(g)$, то $b_0 \sqsubseteq b$.
  - Второ, ясно е, че

$$\begin{aligned} a &= f(a,b) && \text{// } a \text{ е решение на } \bigstar \\ &\sqsupseteq f(a,b_0) && \text{// } b \sqsupseteq b_0 \\ &= \hat{f}(a) && \text{// от деф.} \end{aligned}$$

  Получихме, че $a \in \texttt{Pref}(\hat{f})$. От *Твърдение* 1.11 знаем, че

$$a_0 \stackrel{\text{деф}}{=} \texttt{lfp}(\hat{f}) \sqsubseteq a.$$

  Заключваваме, че $\langle a_0, b_0\rangle \sqsubseteq \langle a, b\rangle$.

□

Нещата започнаха да стават прекалено абстрактни, затова нека да видим един прост пример, който показва, че всъщност горното твърдение е близо до нашата интуиция.

Пример 1.5. Нека да разгледаме следната програма на езика хаскел:

```
ghci> let g(x,y) = if x == 0 then 0 else g(x-1,y) + y
ghci> let f(x) = if x == 0 then 1 else g(x,f(x-1))
```

Да не забравяме, че в хаскел имаме и константата undefined. Това означава, че ако се ограничим до $\mathbb{N}_\perp$, то по горния начин сме дефинирали следните две функции:

$$f(x) = \begin{cases} x!, & \text{ако } x \in \mathbb{N} \\ \perp, & \text{ако } x = \perp \end{cases} \tag{1.3}$$

$$g(x,y) = \begin{cases} 0, & \text{ако } x = 0 \\ x * y, & \text{ако } x, y \in \mathbb{N} \\ \perp, & \text{иначе.} \end{cases} \tag{1.4}$$

Ясно е, че тези функции са монотонни, а следователно и непрекъснати. Целта на *Глава* 3 е да формализираме разсъжданията, които направихме по-горе. Ще видим, че на тази програма можем да съпоставим система от *непрекъснати* изображения.

$x + \perp \stackrel{\text{деф}}{=} \perp$

В *Глава* 3 ще видим, че на всяка програма съпоставяме система от *непрекъснати* изображения. В конкретния пример можем директно

$$\Gamma(f,g)(x) = \begin{cases} 1, & x = 0 \\ g(x, f(\texttt{(-)}(x,1))), & x > 0 \\ \perp, & x = \perp \end{cases}$$

$$\Delta(g)(x,y) = \begin{cases} 0, & x = 0 \\ \texttt{(+)}(g(\texttt{(-)}(x,1),y),y), & x > 0 \\ \perp, & x = \perp. \end{cases}$$

Да видим как можем да дефинираме тези изображения на haskell и как можем получим редицата от апроксимации на най-малките неподвижни точки по Теоремата на Клини.

```
ghci> let gamma(f, g)(x) = if x == 0 then 1 else g(x, f(x - 1))
ghci> let delta(g)(x, y) = if x == 0 then 0 else g(x - 1, y) + y
-- Започваме да строим редицата от апроксимации по Теоремата на Клини
ghci> let g1 = delta( \(x,y) -> undefined )
ghci> let g2 = delta(g1)
ghci> let g3 = delta(g2)
ghci> g3(2,4)
8
ghci> g3(3,4)
*** Exception: Prelude.undefined
-- Можем да подходим и по-мързеливо, като направо дефинираме безкрайния
-- списък от тези апроксимации.
ghci> let approx = (\(x,y) -> undefined):[delta(g) | g <- approx]
ghci> let g9 = approx !! 9
ghci> g9(8,100)
800
ghci> g9(9,100)
*** Exception: Prelude.undefined
-- най-малката неподвижна точка на delta
ghci> let psi(x,y) = (approx !! (x+1))(x,y)
ghci> psi(0,undefined)
0
ghci> psi(2,3)
6
ghci> psi(undefined,0)
*** Exception: Prelude.undefined
```

Горният пример ни подсказва, че с индукция по k, можем да докажем, че ако имаме редицата

$$\begin{aligned} g_0 &= \perp^{(2)} \\ g_{k+1} &= \Delta(g_k), \end{aligned}$$

то, за произволнен индекс $k \geq 1$, имаме

$$g_k(x, y) = \begin{cases} 0, & \text{ако } x = 0 \\ x * y, & \text{ако } x < k \text{ и } y \in \mathbb{N} \\ \bot, & \text{иначе.} \end{cases}$$

Тогава с помощта на Теоремата на Клини можем да докажем, че

$$\mathtt{lfp}(\Delta)(x, y) = \begin{cases} 0, & \text{ако } x = 0 \\ x * y, & \text{ако } x, y \in \mathbb{N} \\ \bot, & \text{ако } \bot \in \{x, y\}. \end{cases}$$

Нека сега да разгледаме изображението

$$\hat{\Gamma}(f)(x) \stackrel{\text{деф}}{=} \Gamma(f, \mathtt{lfp}(\Delta))(x) = \begin{cases} 1, & \text{ако } x = 0 \\ x * f(x-1), & \text{ако } x > 0 \\ \bot, & \text{ако } x = \bot. \end{cases}$$

Нека отново да видим как можем да дефинираме това изображение на хаскел и как можем получим редицата от апроксимации на най-малките неподвижни точки по Теоремата на Клини.

```
ghci> let gamma(f,g)(x) = if x == 0 then 1 else g(x,f(x-1))
ghci> let gamma'(f) = gamma(f, psi)
ghci> :t gamma'
gamma' :: (a -> a) -> a -> a
ghci> let approx' = (\x -> undefined):[gamma'(f) | f <- approx']
ghci> let f9 = approx' !! 9
ghci> f9(8)
40320
ghci> f9(9)
*** Exception: Prelude.undefined
ghci> let phi(x) = (approx' !! (x+1))(x)
ghci> phi(8)      -- phi е най-малката неподвижна точка на gamma'
40320         -- лесно се съобразява, че phi(x) == x!
```

Горният пример ни подсказва, че с индукция по k, можем да докажем, че ако имаме редицата

$$\begin{aligned} f_0 &= \bot^{(1)} \\ f_{k+1} &= \hat{\Gamma}(f_k), \end{aligned}$$

то, за произволен индекс k, имаме

$$f_k(x) = \begin{cases} x!, & \text{ако } x < k \\ \bot, & \text{иначе.} \end{cases}$$

Отново по Теоремата на Клини,

$$\texttt{lfp}(\hat{\Gamma})(x) = \begin{cases} x!, & \text{ако } x \in \mathbb{N} \\ \bot, & \text{ако } x = \bot. \end{cases}$$

От *Твърдение* 1.13 знаем, че двойката $(\texttt{lfp}(\hat{\Gamma})), \texttt{lfp}(\Delta))$ е най-малкото решение на системата, което е точно двойката изображения (f, g) с дефиниции (1.3) и (1.4).

Твърдение 1.14. Да разгледаме изображенията $f \in [\mathcal{A} \xrightarrow{\text{н}} \mathcal{B}]$ и $g \in [\mathcal{A} \xrightarrow{\text{н}} \mathcal{A}]$ и системата от две уравнения:

$$\bigstar = \begin{cases} \quad f(y) = x \\ \quad g(y) = y. \end{cases}$$

Нека $a_0 = \texttt{lfp}(g)$. Тогава най-малкото решение на системата $\bigstar$ е наредената двойка

$$\langle f(a_0), a_0 \rangle.$$

Доказателство.

- Лесно се съобразява, че $\langle f(a_0), a_0 \rangle$ е решение на системата $\bigstar$.
- Нека $\langle c, d \rangle$ е решение на системата $\bigstar$. Тогава $g(d) = d$ и понеже $a_0 = \texttt{lfp}(g)$, то $a_0 \sqsubseteq d$. Освен това, $c = f(d) \sqsupseteq f(a_0)$. Получихме, че $\langle f(a_0), a_0 \rangle \sqsubseteq \langle c, d \rangle$.

Заключаваме, че $\langle f(a_0), a_0 \rangle$ е най-малкото решение на системата $\bigstar$. $\square$

Пример 1.6. Да разгледаме следната програма:

```
ghci> :{  -- използване на multiline дефиниции
ghci> let g(x, y, z) = if x == y + z then z
ghci|                      else if z == x + 1 then 0
ghci|                        else g(x, y, z + 1)
ghci| :}
ghci> let f(x, y) = g(x, y, 0)
```

Лесно се съобразява, че

$$g(x, y) = \begin{cases} x - y, & \text{ако } x \geq y \\ 0, & \text{ако } x < y. \end{cases}$$

Тази функция ще я означаваме като $x \dot{-} y$. На горната програма можем да съпоставим системата от непрекъснати изображения:

$$
\Gamma(g)(x,y) = g(x,y,0)
$$

$$
\Delta(g)(x,y,z) = \begin{cases} z, & \text{ако } x = y + z \\ 0, & \text{ако } z = x + 1 \\ g(x,y,z+1), & \text{иначе и } x,y,z \in \mathbb{N} \\ \bot, & \bot \in \{x,y,z\}. \end{cases}
$$

```
ghci> :{  -- Multiline
ghci> let delta(g)(x, y, z) = if x == y + z then z
ghci|                          else if z == x + 1 then 0
ghci|                            else g(x, y, z + 1)
ghci| :}
ghci> :t delta
delta :: ((t, t, t) -> t) -> (t, t, t) -> t
ghci> let approx = (\(x,y,z) -> undefined):[delta(g) | g <- approx]
ghci> let g9 = approx !! 9
ghci> g9(20,11,1)   -- 20-11 \in [1, 10)
9
ghci> g9(20,1,11) -- 20-1 \in [11, 20)
19
ghci> g9(2,11,4)  -- 2+1 \not\in [4, 13)
*** Exception: Prelude.undefined
```

Горният пример ни подсказва, че с индукция по k, можем да докажем, че ако имаме редицата

$$
\begin{aligned}
g_0 &= \bot^{(3)} \\
g_{k+1} &= \Delta(g_k),
\end{aligned}
$$

то, за произволно k, имаме

$$
g_k(x,y,z) = \begin{cases} 0, & x+1 \in [z, z+k) \\ x-y, & x \geq y \;\&\; x-y \in [z, z+k) \\ \bot, & \text{иначе}. \end{cases}
$$

Тогава можем да приложим Теоремата на Клини и да докажем, че

$$
\mathtt{lfp}(\Delta)(x,y,z) = \begin{cases} x \dot{-} y, & z \leq x+1 \\ \bot, & z > x+1 \text{ или } \bot \in \{x,y,z\}. \end{cases}
$$

Тогава от *Твърдение* 1.14 следва, че

$$
\mathtt{lfp}(\Gamma)(x,y) = \mathtt{lfp}(\Delta)(x,y,0) = \begin{cases} x \dot{-} y, & x,y \in \mathbb{N} \\ \bot, & \bot \in \{x,y\}. \end{cases}
$$

Съобразете, че $\mathtt{lfp}(\Gamma) = \bigsqcup_k f_k$, където $f_k(x,y) = g_k(x,y,0)$.

Лема 1.3. Нека $f \in [\mathcal{A} \times \mathcal{B} \overset{\text{н}}{\to} \mathcal{A}]$. За $f_a(y) \overset{\text{деф}}{=} f(a, y)$ знаем, че $f_a \in [\mathcal{B} \overset{\text{н}}{\to} \mathcal{B}]$. Да положим $\boldsymbol{b}(a) \overset{\text{деф}}{=} \mathtt{lfp}(f_a)$. Докажете, че $\boldsymbol{b}(x) \in [\mathcal{A} \overset{\text{н}}{\to} \mathcal{B}]$.

Теорема 1.8 (Бекич). Нека $f \in [\mathcal{A} \times \mathcal{B} \overset{\text{н}}{\to} \mathcal{A}]$ и $g \in [\mathcal{A} \times \mathcal{B} \overset{\text{н}}{\to} \mathcal{B}]$. Да разгледаме системата

$$\left|\begin{array}{lcl} f(x, y) & = & x \\ g(x, y) & = & y. \end{array}\right.$$

Да положим

$$\left|\begin{array}{lcl} \hat{f}(a) & \overset{\text{деф}}{=} & f(a, \boldsymbol{b}(a)) \\ \hat{g}(b) & \overset{\text{деф}}{=} & g(\boldsymbol{a}(b), b). \end{array}\right.$$

Докажете, че:

(а) $(\boldsymbol{a}(\mathtt{lfp}(\hat{g})), \mathtt{lfp}(\hat{g}))$ е най-малкото решение на системата.

(б) $(\mathtt{lfp}(\hat{f}), \mathtt{lfp}(\hat{g}))$ е най-малкото решение на системата.

1.9 Изоморфни области на Скот

[12, стр. 124].

Нека $\mathcal{A}_1 = (A_1, \sqsubseteq_1, \perp_1)$ и $\mathcal{A}_2 = (A_2, \sqsubseteq_2, \perp_2)$ са области на Скот. Ще казваме, че $\mathcal{A}_1$ е **изоморфна** на $\mathcal{A}_2$, което ще означаваме като

$$\mathcal{A}_1 \cong \mathcal{A}_2,$$

ако съществува *биективна* функция $F : A_1 \to A_2$ със свойството:

$$(\forall a \in A_1)(\forall b \in A_1)[\, a \sqsubseteq_1 b \iff F(a) \sqsubseteq_2 F(b)\,].$$

В такъв случай ще казваме, че F задава изоморфизъм между $\mathcal{A}_1$ и $\mathcal{A}_2$.

Когато искаме да означим, че $\mathcal{A}_1$ е изоморфна на $\mathcal{A}_2$ чрез F, то понякога ще пишем $\mathcal{A}_1 \cong_F \mathcal{A}_2$.

Задача 1.27. Докажете, че ако $\mathcal{A}_1 \cong_F \mathcal{A}_2$, то $F(\perp_1) = \perp_2$.

Твърдение 1.15. Ако $\mathcal{A}_1 \cong_F \mathcal{A}_2$, то $F \in [\mathcal{A}_1 \stackrel{\text{н}}{\to} \mathcal{A}_2]$.

Задача в Кеймбридж 2022 г. [6].

Упътване. Да разгледаме произволна верига $(a_i)_{i=0}^{\infty}$ от елементи на $\mathcal{A}_1$. Ще докажем, че

$$F(\bigsqcup_i a_i) = \bigsqcup_i F(a_i).$$

- Първо, от дефиницията веднага следва, че F е монотонно изображение, защото
$$a \sqsubseteq_1 b \implies F(a) \sqsubseteq_2 F(b).$$
Това означава, че $(F(a_i))_{i=0}^{\infty}$ е монотонно растяща верига от елементи на $\mathcal{A}_2$. От *Твърдение* 1.4 получаваме, че
$$\bigsqcup_i F(a_i) \sqsubseteq_2 F(\bigsqcup_i a_i).$$

- За другата посока, нека $b \in \mathcal{A}_2$ е горна граница на веригата $(F(a_i))_{i=0}^{\infty}$, т.е.
$$(\forall i)[\, F(a_i) \sqsubseteq_2 b\,].$$
Ще докажем, че $F(\bigsqcup_i a_i) \sqsubseteq_2 b$. Понеже F е *върху* A_2, то съществува елемент $a \in A_1$, такъв че $F(a) = b$. Тогава:
$$\begin{aligned}(\forall i)[\, F(a_i) \sqsubseteq_2 F(a)\,] &\implies (\forall i)[\, a_i \sqsubseteq_1 a\,] && \text{// } F \text{ е изоморфизъм}\\ &\implies \bigsqcup_i a_i \sqsubseteq_1 a && \text{// } a \text{ е горна граница}\\ &\implies F(\bigsqcup_i a_i) \sqsubseteq_1 F(a). && \text{// } F \text{ е изоморфизъм}\end{aligned}$$
Понеже $b = F(a)$, заключаваме, че
$$F(\bigsqcup_i a_i) \sqsubseteq_2 b.$$

$\square$

Твърдение 1.16. Нека $f \in [\mathcal{A}_1 \overset{\text{м}}{\to} \mathcal{A}_2]$ и $g \in [\mathcal{A}_2 \overset{\text{м}}{\to} \mathcal{A}_1]$, като

- $f \circ g = \mathtt{id}_2$;
- $g \circ f = \mathtt{id}_1$.

$\mathtt{id}_i(a) \overset{\text{деф}}{=} a$ за вс. $a \in \mathcal{A}_i$

Тогава са изпълнени свойствата:

(1) $\mathcal{A}_1 \cong_f \mathcal{A}_2$;

(2) $\mathcal{A}_2 \cong_g \mathcal{A}_1$;

Твърдение 1.17. Нека $\mathcal{A}_1 \cong_F \mathcal{A}_2$. Тогава:

(1) $[\mathcal{A}_1 \overset{\text{н}}{\to} \mathcal{A}_1] \cong_G [\mathcal{A}_2 \overset{\text{н}}{\to} \mathcal{A}_2]$, където

$$G(f) \overset{\text{деф}}{=} F \circ f \circ F^{-1};$$

Графично това може да се изобрази така:

$$\begin{tikzcd}
\mathcal{A}_1 \arrow[r, "f"] \arrow[d, leftarrow, "F^{-1}"'] & \mathcal{A}_1 \arrow[d, "F"] \\
\mathcal{A}_2 \arrow[r, dashed, "G(f)"] & \mathcal{A}_2
\end{tikzcd}$$

(2) ако $f \in [\mathcal{A}_1 \overset{\text{н}}{\to} \mathcal{A}_1]$, то

$$F(\mathtt{lfp}(f)) = \mathtt{lfp}(G(f)).$$

Упътване. Ще докажем (1) като използвме *Твърдение* 1.16.

- Ще докажем, че G е монотонно изображение. Нека $f, h \in [\mathcal{A}_1 \overset{\text{н}}{\to} \mathcal{A}_1]$ и $f \sqsubseteq h$, т.е.
$$(\forall a \in \mathcal{A}_1)[\, f(a) \sqsubseteq_1 h(a)\,].$$
Ще докажем, че $G(f) \sqsubseteq G(h)$, т.е.
$$(\forall b \in \mathcal{A}_2)[\, G(f)(b) \sqsubseteq_1 G(h)(b)\,].$$
Да разгледаме произволен елемент $b \in \mathcal{A}_2$. Понеже F е биекция, то съществува елемент $a \in A_1$, такъв че $F(a) = b$, т.е. $F^{-1}(b) = a$. Тогава:
$$\begin{aligned}
G(f)(b) &\overset{\text{деф}}{=} F(f(F^{-1}(b))) \\
&= F(f(a)) && /\!/\ F^{-1}(b) = a \\
&\sqsubseteq_2 F(h(a)) && /\!/\ f(a) \sqsubseteq h(a) \text{ и } F \text{ е изом.} \\
&= F(h(F^{-1}(b))) && /\!/\ F^{-1}(b) = a \\
&\overset{\text{деф}}{=} G(h)(b).
\end{aligned}$$

- Нека $G(f) \sqsubseteq G(h)$. Ще докажем, че $f \sqsubseteq h$. За целта, нека $a \in A_1$. Понеже F е сюрективна, то съществува $b \in A_2$, за който $F^{-1}(b) = a$. Понеже

$$G(f) \stackrel{\text{деф}}{=} F \circ f \circ F^{-1} \sqsubseteq F \circ h \circ F^{-1} = G(h),$$

то получаваме, че

$$F(f(F^{-1}(b))) \sqsubseteq_2 F(h(F^{-1}(b))).$$

Оттук,

$$F(f(a)) \sqsubseteq_2 F(h(a)) \implies f(a) \sqsubseteq_1 h(a),$$

защото F е изоморфизъм.

Сега преминаваме към доказателството на (2). Понеже f е непрекъснато изображение е ясно, че $(f^n(\bot_1))_{n=0}^{\infty}$ е верига. Също така знаем, че

$$\mathtt{lfp}(f) = \bigsqcup_n f^n(\bot_1).$$

Да напомним, че за $f \in [\mathcal{A}_1 \stackrel{\text{н}}{\to} \mathcal{A}_1]$, означаваме

$$f^0 = \lambda x.\bot_1$$
$$f^{n+1} = f \circ f^n.$$

След аналогични разсъждения можем да съобразим, че

$$\mathtt{lfp}(G(f)) = \bigsqcup_n G(f)^n(\bot_2).$$

Първо ще докажем с индукция по n, че

$$(\forall n)[\,(G(f))^n = G(f^n)\,]. \tag{1.5}$$

- За $n = 0$ имаме, че за произволен елемент $b \in \mathcal{A}_2$,

$$\begin{aligned}
(G(f))^0(b) &\stackrel{\text{деф}}{=} \bot_2 \\
&= F(\bot_1) && \text{// } F \text{ е изом.} \\
&= F(f^0(F^{-1}(b))) && \text{// } f^0(F^{-1}(b)) \stackrel{\text{деф}}{=} \bot_1 \\
&= (F \circ f^0 \circ F^{-1})(b) \\
&\stackrel{\text{деф}}{=} G(f^0)(b).
\end{aligned}$$

- Нека да приемем, че твърдението е вярно за n. Тогава за $n+1$ имаме, че:

$$\begin{aligned}
(G(f))^{n+1} &\stackrel{\text{деф}}{=} G(f) \circ (G(f))^n \\
&= G(f) \circ G(f^n) && \text{// от И.П.} \\
&\stackrel{\text{деф}}{=} (F \circ f \circ F^{-1}) \circ (F \circ f^n \circ F^{-1}) \\
&= F \circ f \circ (F^{-1} \circ F) \circ f^n \circ F^{-1} \\
&= F \circ f \circ f^n \circ F^{-1} && \text{// } F^{-1} \circ F = id \\
&= F \circ f^{n+1} \circ F^{-1} && \text{// } f \circ f^n = f^{n+1} \\
&\stackrel{\text{деф}}{=} G(f^{n+1}).
\end{aligned}$$

Тогава:

$$
\begin{aligned}
F(\texttt{lfp}(f)) &= F(\bigsqcup_n f^n(\bot_1)) && // \ \texttt{lfp}(f) = \bigsqcup_n f^n(\bot_1) \\
&= \bigsqcup_n F(f^n(\bot_1)) && // \ F \text{ е непр.} \\
&= \bigsqcup_n F(f^n(F^{-1}(\bot_2))) && // \ F^{-1}(\bot_2) = \bot_1 \\
&= \bigsqcup_n (F \circ f^n \circ F^{-1})(\bot_2) \\
&\stackrel{\text{деф}}{=} \bigsqcup_n G(f^n)(\bot_2) \\
&= \bigsqcup_n G(f)^n(\bot_2) && // \text{ от } (1.5) \\
&= \texttt{lfp}(G(f)).
\end{aligned}
$$

□

Твърдение 1.18. За произволни области на Скот $\mathcal{A}$, $\mathcal{B}$ и $\mathcal{C}$ е изпълнено, че

$$[\mathcal{A} \xrightarrow{\text{н}} [\mathcal{B} \xrightarrow{\text{н}} \mathcal{C}]] \cong [\mathcal{A} \times \mathcal{B} \xrightarrow{\text{н}} \mathcal{C}].$$

[12, стр. 139].

Упътване. Докажете, че изображението

$$\texttt{curry} : [\mathcal{A} \times \mathcal{B} \xrightarrow{\text{н}} \mathcal{C}] \to [\mathcal{A} \xrightarrow{\text{н}} [\mathcal{B} \xrightarrow{\text{н}} \mathcal{C}]],$$

където

$$\texttt{curry}(f)(a)(b) \stackrel{\text{деф}}{=} f(a, b)$$

задава изоморфизъм. □

Когато на хаскел пишем типовата сигнатура на някоя функция като

```
f :: a -> b -> c
```

в действителност се има предвид

```
f :: a -> (b -> c)
```

На практика тези две задачи ни казват, че няма значение дали използваме *curried* или *uncurried* версията на една функция. На `хаскел` е по-удобно да използваме *curried* версията, защото като фиксираме първия аргумент на една функция получаваме нова функция наготово. Например,

```
ghci> let plus1 = (+) 1
ghci> :t plus1
plus1 :: Num a => a -> a
ghci> plus1 2
3
```

Това се нарича *partial application*. Вижте https://wiki.haskell.org/Partial_application.

Нека да дефинираме

$$\emptyset_\perp \stackrel{\text{деф}}{=} (\{\perp\}, \sqsubseteq, \perp).$$

Задача 1.28. Докажете, че за произволна област на Скот $\mathcal{A}$ е изпълнено:

$$[\emptyset_\perp \xrightarrow{\text{н}} \mathcal{A}] \cong \mathcal{A}$$
$$[\mathcal{A} \xrightarrow{\text{н}} \emptyset_\perp] \cong \emptyset_\perp.$$

Задача 1.29. Докажете, че съществуват области на Скот $\mathcal{A}$, $\mathcal{B}$ и $\mathcal{C}$, за които

$$[[\mathcal{A} \xrightarrow{\text{н}} \mathcal{B}] \xrightarrow{\text{н}} \mathcal{C}] \ncong [\mathcal{A} \xrightarrow{\text{н}} [\mathcal{B} \xrightarrow{\text{н}} \mathcal{C}]].$$

Упътване. Нека изберем $\mathcal{A} = \mathcal{C} = \mathbb{N}_\perp$, а $\mathcal{B} = \emptyset_\perp$. Тогава

$$[[\mathbb{N}_\perp \xrightarrow{\text{н}} \emptyset_\perp] \xrightarrow{\text{н}} \mathbb{N}_\perp] \cong [\emptyset_\perp \xrightarrow{\text{н}} \mathbb{N}_\perp] \cong \mathbb{N}_\perp,$$

но лесно можем да съобразим, че:

$$[\mathbb{N}_\perp \xrightarrow{\text{н}} [\emptyset_\perp \xrightarrow{\text{н}} \mathbb{N}_\perp]] \cong [\mathbb{N}_\perp \xrightarrow{\text{н}} \mathbb{N}_\perp].$$

Сега остава да съобразим, че

$$\mathbb{N}_\perp \ncong [\mathbb{N}_\perp \xrightarrow{\text{н}} \mathbb{N}_\perp].$$

□

Глава 2

Езикът REG

2.1 Регулярни изрази

Следната абстрактна граматика

$$\tau ::= \varepsilon \mid a \cdot X \mid \tau + \tau.$$

описва термовете на езика REG , където a означава буква, която принадлежи на някаква предварително зададена азбука Σ, а X означава променлива. Нека имаме и безкраен набор от променливи $X_0, X_1, \ldots$. Програмите на езика REG имат следния вид:

Обърнете внимание, че нямаме звездата на Клини в дефиницията на термовете.

Ще използваме малките гръцки букви τ, ρ, μ, евентуално с индекси, за да означаваме термовете. Думите над азбуката Σ ще означаваме с малките гръцки букви α, β, γ.

$$\left|\begin{array}{lcl} X_1 & = & \tau_1[X_1, \ldots, X_n] \\ & \vdots & \\ X_n & = & \tau_n[X_1, \ldots, X_n]. \end{array}\right.$$

Ето един пример за такава програма:

$$\begin{aligned} X_1 &= \overbrace{a \cdot X_1 + b \cdot X_2}^{\tau_1[X_1, X_2]} \\ X_2 &= \underbrace{b \cdot X_1 + \varepsilon}_{\tau_2[X_1, X_2]}. \end{aligned}$$

В тази глава ще видим, че регулярните изрази могат да се опишат като програми на езика REG . Ще разгледаме два начина как точно една програма може да породи регулярен език.

2.2 Операционна семантика

Да фиксираме една програма R на езика REG :

$$\left|\begin{array}{lcl} X_1 & = & \tau_1[X_1,\dots,X_n] \\ & \vdots & \\ X_n & = & \tau_n[X_1,\dots,X_n]. \end{array}\right.$$

Тук е важно да се отбележи, че дефиницията на операционната семантика не можем да я дадем с индукция по построението на изрази. Проблемът е в случая (rec). Също така, няма как да дефинираме релация $\tau \Downarrow_{\mathtt{R}} L$, където L е език, защото все пак искаме всяко изчисление да трае краен брой стъпки, а това няма как да стане, когато L е безкраен език.

Щом сме фиксирали програмата R, с която искаме да работим, това означава, че вече знаем и буквите, които се срещат в програмата, а също и променливите, които са $X_1,\dots,X_n$. С индукция по ℓ, дефинираме релацията $\tau \Downarrow_{\mathtt{R}}^{\ell} \alpha$, където τ е терм на езика REG със свободни променливи измежду $X_1,\dots,X_n$.

$$\text{(eps)}\ \frac{}{\varepsilon \Downarrow_{\mathtt{R}}^{0} \varepsilon} \qquad\qquad \frac{\tau_i \Downarrow_{\mathtt{R}}^{\ell} \beta \quad \alpha = a\cdot\beta}{a\cdot X_i \Downarrow_{\mathtt{R}}^{\ell+1} \alpha}\ \text{(rec)}$$

$$\text{(left-or)}\ \frac{\rho \Downarrow_{\mathtt{R}}^{\ell} \alpha}{\rho+\mu \Downarrow_{\mathtt{R}}^{\ell+1} \alpha} \qquad\qquad \frac{\mu \Downarrow_{\mathtt{R}}^{\ell} \alpha}{\rho+\mu \Downarrow_{\mathtt{R}}^{\ell+1} \alpha}\ \text{(right-or)}$$

Фигура 2.1: Правила на операционната семантика на езика REG .

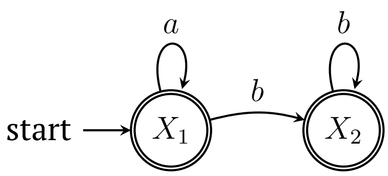
Ще пишем $\tau \Downarrow_{\mathtt{R}} \alpha$ точно тогава, когато съществува ℓ, за което $\tau \Downarrow_{\mathtt{R}}^{\ell} \alpha$.

На всяка променлива от програмата R съответства състояние на автомата.

Пример 2.1. Нека $\Sigma = \{a,b\}$. Да разгледаме програмата R, където:

$$\begin{aligned} X_1 &= \overbrace{a\cdot X_1 + b\cdot X_2 + \varepsilon}^{\tau_1[X_1,X_2]} \\ X_2 &= \underbrace{b\cdot X_2 + \varepsilon}_{\tau_2[X_1,X_2]}. \end{aligned}$$

Съобразете, че за всеки две естествени числа n и k е изпълнено, че $\tau_1 \Downarrow_{\mathtt{R}} a^k b^n$ и $\tau_2 \Downarrow_{\mathtt{R}} b^n$.



Фигура 2.2: Краен автомат съотвестващ на програмата R.

2.3 Денотационна семантика

Тук отново приемаме, че сме работим с предварително фиксирана програма какво в Раздел 2.2.

Задача 2.1. Докажете, че $\mathcal{S} = (\mathcal{P}(\Sigma^\star), \subseteq, \emptyset)$ е област на Скот.

За всеки терм $\tau[X_1, \ldots, X_n]$ дефинираме изображението

$$[\![\tau]\!] : \mathcal{P}(\Sigma^\star)^n \to \mathcal{P}(\Sigma^\star)$$

по следния начин:

- $[\![\varepsilon]\!](L_1, \ldots, L_n) = \{\varepsilon\}$.
- $[\![a \cdot X_j]\!](L_1, \ldots, L_n) = \{a\} \cdot L_j$.
- $[\![\rho + \mu]\!](L_1, \ldots, L_n) = [\![\rho]\!](L_1, \ldots, L_n) \cup [\![\mu]\!](L_1, \ldots, L_n)$.

Задача 2.2. Докажете, че за всеки терм τ на езика REG имаме, че $[\![\tau]\!]$ е непрекъснато изображение в областта на Скот $\mathcal{S} = (\mathcal{P}(\Sigma^\star), \subseteq, \emptyset)$.

Програмата е просто редове с текст. На този текст съответства система от уравнения от непрекъснати изображения.

Пример 2.2. Да разгледаме програмата R, където:

$$\begin{aligned} X_1 &= \overbrace{b \cdot X_1 + a \cdot X_2}^{\tau_1[X_1, X_2]} \\ X_2 &= \underbrace{\varepsilon}_{\tau_2[X_1, X_2]} \end{aligned}$$

Според горната дефиниция на семантика на термовете, на програмата R съответства следната система от непрекъснати изображения:

$$\begin{aligned} L &= \overbrace{\{b\} \cdot L \cup \{a\} \cdot M}^{[\![\tau_1]\!](L, M)} \\ M &= \underbrace{\{\varepsilon\}}_{[\![\tau_2]\!](L, M)} \end{aligned}$$

От *Теорема* 1.7 знаем, че тази система има най-малко решение. Нека да го намерим. За тази цел, да дефинираме непрекъснатото изображение $\Gamma : \mathcal{P}(\Sigma^\star)^2 \to \mathcal{P}(\Sigma^\star)^2$ като

От *Задача* 1.11 знаем, че Γ е непрекъснато изображение.

$$\Gamma \stackrel{\text{деф}}{=} [\![\tau_1]\!] \times [\![\tau_2]\!],$$

т.е. за произволни езици L и M над азбуката Σ е изпълнено, че:

$$\Gamma(L, M) = ([\![\tau_1]\!](L, M), [\![\tau_2]\!](L, M)).$$

От Теоремата на Клини знаем как можем да намерим най-малката неподвижна точка на Γ, която ще бъде и най-малкото решение на горната система.

- $(L_0, M_0) \stackrel{\text{деф}}{=} (\emptyset, \emptyset)$;

- $(L_1, M_1) \stackrel{\text{деф}}{=} \Gamma(L_0, M_0) = (\llbracket\tau_1\rrbracket(L_0, M_0), \llbracket\tau_2\rrbracket(L_0, M_0)) = (\emptyset, \{\varepsilon\})$;
- $(L_2, M_2) \stackrel{\text{деф}}{=} \Gamma(L_1, M_1) = (\llbracket\tau_1\rrbracket(L_1, M_1), \llbracket\tau_2\rrbracket(L_1, M_1)) = (\{a\}, \{\varepsilon\})$;
- $(L_3, M_3) \stackrel{\text{деф}}{=} \Gamma(L_2, M_2) = (\llbracket\tau_1\rrbracket(L_2, M_2), \llbracket\tau_2\rrbracket(L_2, M_2)) = (\{ba, a\}, \{\varepsilon\})$;
- $(L_4, M_4) \stackrel{\text{деф}}{=} \Gamma(L_3, M_3) = (\llbracket\tau_1\rrbracket(L_3, M_3), \llbracket\tau_2\rrbracket(L_3, M_3)) = (\{bba, ba, a\}, \{\varepsilon\})$;
- $(L_5, M_5) \stackrel{\text{деф}}{=} \Gamma(L_4, M_4) = (\llbracket\tau_1\rrbracket(L_4, M_4), \llbracket\tau_2\rrbracket(L_4, M_4)) = (\{bba, bba, ba, a\}, \{\varepsilon\})$.

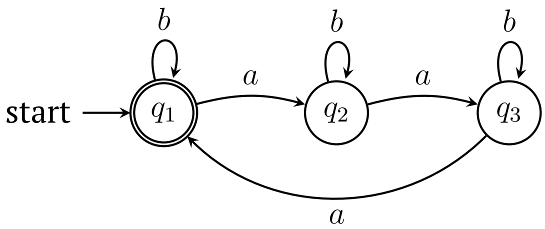
✍ Докажете!

Лесно се съобразява, че $L_n = \{b^k a \mid k < n-1\}$, а $M_n = \{\varepsilon\}$. Тогава

$$\mathtt{lfp}(\Gamma) = (\bigcup_n L_n, \bigcup_n M_n) = (\{b\}^\star \cdot \{a\}, \{\varepsilon\}).$$

Пример 2.3. Да разгледаме следния краен детерминиран автомат:

Означаваме с $|\omega|_a$ броя на срещанията на a в думата ω.



Фигура 2.3: Автомат разпознаващ $\{\omega \in \{a,b\}^\star \mid |\omega|_a \equiv 0 \mod 3\}$

На този автомат съответства следната програма на езика REG :

$$\left|\begin{array}{lcl} X_1 & = & a \cdot X_2 + b \cdot X_1 + \varepsilon \\ X_2 & = & a \cdot X_3 + b \cdot X_2 \\ X_3 & = & a \cdot X_1 + b \cdot X_3. \end{array}\right.$$

На тази програма съответства системата от уравнения:

$$\left|\begin{array}{lcl} L_1 & = & \{a\} \cdot L_2 \cup \{b\} \cdot L_1 \cup \{\varepsilon\} \\ L_2 & = & \{a\} \cdot L_3 \cup \{b\} \cdot L_2 \\ L_3 & = & \{a\} \cdot L_1 \cup \{b\} \cdot L_3. \end{array}\right.$$

Най-малкото решение на тази система е тройката $(\hat{L}_1, \hat{L}_2, \hat{L}_3)$ от следните езици:

✍ Докажете!

$$\begin{aligned} \hat{L}_1 &= \{\omega \in \{a,b\}^\star \mid |\omega|_a \equiv 0 \mod 3\} \\ \hat{L}_2 &= \{\omega \in \{a,b\}^\star \mid |\omega|_a \equiv 2 \mod 3\} \\ \hat{L}_3 &= \{\omega \in \{a,b\}^\star \mid |\omega|_a \equiv 1 \mod 3\}. \end{aligned}$$

Пример 2.4 ни подсказва защо се интересуваме от най-малкото решение на една система, а не просто от някое решение на сисатемата. ✍ Какъв краен автомат съответства на системата от *Пример* 2.4?

Пример 2.4. Да разгледаме следната система съставена само от едно уравнение:

$$L = \{\varepsilon, a\} \cdot L.$$

Най-малкото решение на тази система е езикът $\emptyset$, но тази система има за решение и езикът $\{a^n \mid n \geq 0\} = \{a\}^\star$.

Задача 2.3. Докажете, че най-малкото решение на системата

$$\left| \begin{array}{lcl} X_1 & = & a \cdot X_1 + b \cdot X_2 + \varepsilon \\ X_2 & = & b \cdot X_2 + \varepsilon \end{array} \right.$$

е двойката от езици $(\{a\}^\star \cdot \{b\}^\star, \{b\}^\star)$.

Задача 2.4 (Теорема за характеризация)**.** Да разгледаме системата от непрекъснати оператори

$$\left| \begin{array}{lcl} L_1 & = & [\![\tau_1]\!](L_1, \dots, L_n) \\ & \vdots & \\ L_n & = & [\![\tau_n]\!](L_1, \dots, L_n). \end{array} \right.$$

Знаем, че тя притежава най-малко решение $(\hat{L}_1, \dots, \hat{L}_n)$. Докажете, че всеки от езиците $\hat{L}_i$ е регулярен.

Докажете, че всеки регулярен език е елемент от най-малкото решение на някоя система от непрекъснати изображения от горния вид.

2.4 Теорема за еквивалентност

Нека оттук нататък да фиксираме една програма R на езика REG :

$$
\begin{array}{|lcl}
X_1 & = & \tau_1[X_1, \ldots, X_n] \\
 & \vdots & \\
X_n & = & \tau_n[X_1, \ldots, X_n].
\end{array}
$$

Нека също $(L_1, \ldots, L_n)$ да бъде най-малкото решение на непрекъснатото изображение $\Gamma \stackrel{\text{деф}}{=} [\![\tau_1]\!] \times \cdots \times [\![\tau_n]\!]$. Целта ни е да докажем, че за всяко $i = 1, \ldots, n$ е изпълнено, че:

$$L_i = \{\alpha \in \Sigma^\star \mid \tau_i \Downarrow_{\mathtt{R}} \alpha\}.$$

Този резултат ще наречем теорема за еквивалентност. Това ще направим на две стъпки.

Да напомним, че $L_i = \bigcup_k L_i^k$, където:

$$
\begin{aligned}
& L_i^0 = \emptyset \\
& L_i^{k+1} = [\![\tau_i]\!](L_1^k, \ldots, L_n^k).
\end{aligned}
$$

Лема 2.1. За всеки индекс r и всеки терм τ на езика REG с променливи измежду $X_1, \ldots, X_n$ е изпълнено, че:

$$[\![\tau]\!](L_1^r, \ldots, L_n^r) \subseteq \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}} \alpha\}.$$

Доказателство. Нека да кръстим **Include**(r) твърдението „за всеки терм τ с променливи измежду $X_1, \ldots, X_n$ е изпълнено, че:

$$[\![\tau]\!](L_1^r, \ldots, L_n^r) \subseteq \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}} \alpha\}".$$

Целта ни е да докажем, че $(\forall r \in \mathbb{N})$**Include**$(r)$. Това ще направим с индукция по r.

Нека $r = 0$. Знаем, че $L_i^0 = \emptyset$ за всеки индекс i измежду $1, \ldots, n$. За да докажем, че **Include**(0) е изпълнено, ще направим индукция по построението на термовете τ за да докажем, че:

Тук е важно, че дефиницията на денотационната семантика следва индуктивното построение на термовете.

$$[\![\tau]\!](L_1^0, \ldots, L_n^0) \subseteq \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}} \alpha\} \tag{2.1}$$

- Нека $\tau \equiv \varepsilon$. Тогава $[\![\tau]\!](L_1^0, \ldots, L_n^0) = \{\varepsilon\}$. От правилата на операционната семантика имаме, че $\varepsilon \Downarrow_{\mathtt{R}} \varepsilon$. Заключаваме, че Свойство (2.1) е изпълнено.

- Нека $\tau \equiv a \cdot X_j$, за някой индекс j измежду $1, \ldots, n$. Тогава

$$[\![\tau]\!](L_1^0, \ldots, L_n^0) = \{a\} \cdot L_j^0 = \{a\} \cdot \emptyset = \emptyset$$

и оттук е очевидно, че Свойство (2.1) е в сила.

- Нека $\tau \equiv \rho + \mu$. Тогава $[\![\tau]\!](L_1^0, \ldots, L_n^0) = [\![\rho]\!](L_1^0, \ldots, L_n^0) \cup [\![\mu]\!](L_1^0, \ldots, L_n^0)$. Сега от **(И.П.)** за Свойство (2.1) получаваме, че

$$\begin{aligned}[\![\rho]\!](L_1^0, \ldots, L_n^0) &\subseteq \{\alpha \in \Sigma^\star \mid \rho \Downarrow_{\mathtt{R}} \alpha\}\\ [\![\mu]\!](L_1^0, \ldots, L_n^0) &\subseteq \{\alpha \in \Sigma^\star \mid \mu \Downarrow_{\mathtt{R}} \alpha\}.\end{aligned}$$

  От правилата на операционната семантика имаме, че

$$\text{(left-or)}\ \frac{\rho \Downarrow_{\mathtt{R}} \alpha}{\rho + \mu \Downarrow_{\mathtt{R}} \alpha} \qquad\qquad \frac{\mu \Downarrow_{\mathtt{R}} \alpha}{\rho + \mu \Downarrow_{\mathtt{R}} \alpha}\ \text{(right-or)}$$

  Оттук веднага получаваме, че:

$$\{\alpha \in \Sigma^\star \mid \rho \Downarrow_{\mathtt{R}} \alpha\} \cup \{\alpha \in \Sigma^\star \mid \mu \Downarrow_{\mathtt{R}} \alpha\} = \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}} \alpha\}.$$

  Заключаваме, че:

$$\begin{aligned}[\![\tau]\!](L_1^0, \ldots, L_n^0) &= [\![\rho]\!](L_1^0, \ldots, L_n^0) \cup [\![\mu]\!](L_1^0, \ldots, L_n^0)\\ &\subseteq \{\alpha \in \Sigma^\star \mid \rho \Downarrow_{\mathtt{R}} \alpha\} \cup \{\alpha \in \Sigma^\star \mid \mu \Downarrow_{\mathtt{R}} \alpha\}\\ &= \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}} \alpha\}.\end{aligned}$$

  Свойство 2.1 е изпълнено за всеки терм τ.

Нека $r > 0$ и като индукционно предположение да приемем, че е изпълнено **Include**$(r-1)$. Ще докажем, че е изпълнено **Include**(r). Ще направим индукция по построението на термовете τ за да докажем, че:

Тук знаем, че $L_i^r = [\![\tau_i]\!](L_1^{r-1}, \ldots, L_n^{r-1})$.

$$[\![\tau]\!](L_1^r, \ldots, L_n^r) \subseteq \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}} \alpha\}. \tag{2.2}$$

- Нека $\tau \equiv \varepsilon$. Отново, понеже $[\![\tau]\!](L_1^r, \ldots, L_n^r) = \{\varepsilon\}$, то от правилата на операционната семантика имаме, че $\varepsilon \Downarrow_{\mathtt{R}} \varepsilon$. Заключаваме, че Свойство (2.2) е изпълнено.

- Нека $\tau \equiv a \cdot X_j$ за някой индекс j измежду $1, \ldots, n$. Тогава

$$[\![\tau]\!](L_1^r, \ldots, L_n^r) = \{a\} \cdot L_j^r.$$

  Понеже сме приели, че **Include**$(r-1)$ е изпълнено, то имаме автоматично, че:

$$L_j^r = [\![\tau_j]\!](L_1^{r-1}, \ldots, L_n^{r-1}) \subseteq \{\beta \in \Sigma^\star \mid \tau_j \Downarrow_{\mathtt{R}} \beta\}.$$

  От правилата на операционната семантика имаме, че:

$$\frac{\tau_j \Downarrow_{\mathtt{R}} \beta}{a \cdot X_j \Downarrow_{\mathtt{R}} a \cdot \beta}\ \text{(rec)}$$

Понеже $\tau \equiv a \cdot X_j$, получаваме следното:

$$\{a\} \cdot \{\beta \in \Sigma^\star \mid \tau_j \Downarrow_{\mathtt{R}} \beta\} = \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}} \alpha\}. \tag{2.3}$$

Заключаваме, че:

$$\begin{aligned}
[\![\tau]\!](L_1^r, \dots, L_n^r) &= \{a\} \cdot L_j^r \\
&= \{a\} \cdot [\![\tau_j]\!](L_1^{r-1}, \dots, L_n^{r-1}) \\
&\subseteq \{a\} \cdot \{\beta \in \Sigma^\star \mid \tau_j \Downarrow_{\mathtt{R}} \beta\} && \text{// от } \textbf{Include}(r-1) \\
&= \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}} \alpha\}. && \text{// от Свойство (2.3)}
\end{aligned}$$

- Нека $\tau \equiv \rho + \mu$. Тук имаме, че

$$[\![\tau]\!](L_1^r, \dots, L_n^r) = [\![\rho]\!](L_1^r, \dots, L_n^r) \cup [\![\mu]\!](L_1^r, \dots, L_n^r).$$

Понеже τ е построен с помощта на ρ и μ, можем да използваме (вътрешното) индукционно предположение за ρ и μ, откъдето имаме, че:

$$\begin{aligned}
[\![\rho]\!](L_1^r, \dots, L_n^r) &\subseteq \{\alpha \in \Sigma^\star \mid \rho \Downarrow_{\mathtt{R}} \alpha\} \\
[\![\mu]\!](L_1^r, \dots, L_n^r) &\subseteq \{\alpha \in \Sigma^\star \mid \mu \Downarrow_{\mathtt{R}} \alpha\}.
\end{aligned}$$

От правилата на операционната семантика имаме, че

$$\{\alpha \in \Sigma^\star \mid \rho \Downarrow_{\mathtt{R}} \alpha\} \cup \{\alpha \in \Sigma^\star \mid \mu \Downarrow_{\mathtt{R}} \alpha\} = \{\alpha \in \Sigma^\star \mid \rho + \mu \Downarrow_{\mathtt{R}} \alpha\}.$$

Заключаваме, че:

$$\begin{aligned}
[\![\tau]\!](L_1^r, \dots, L_n^r) &= [\![\rho]\!](L_1^r, \dots, L_n^r) \cup [\![\mu]\!](L_1^r, \dots, L_n^r) \\
&\subseteq \{\alpha \in \Sigma^\star \mid \rho \Downarrow_{\mathtt{R}} \alpha\} \cup \{\alpha \in \Sigma^\star \mid \mu \Downarrow_{\mathtt{R}} \alpha\} \\
&= \{\alpha \in \Sigma^\star \mid \mu + \rho \Downarrow_{\mathtt{R}} \alpha\} \\
&= \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}} \alpha\}.
\end{aligned}$$

Свойство (2.2) е изпълнено за всеки терм τ.

□

Следствие 2.1. Нека $L_1, \dots, L_n$ е най-малкото решение на непрекъснатото изображение $\Gamma \stackrel{\text{деф}}{=} [\![\tau_1]\!] \times \dots \times [\![\tau_n]\!]$. Тогава за произволен терм τ с променливи измежду $X_1, \dots, X_n$ е изпълнено, че:

$$[\![\tau]\!](L_1, \dots, L_n) \subseteq \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}} \alpha\}.$$

Доказателство. Използваме, че $[\![\tau]\!]$ е непрекъснато изображение. Тогава

$$[\![\tau]\!](L_1, \dots, L_n) = [\![\tau]\!](\bigcup_r L_1^r, \dots, \bigcup_r L_n^r)$$

$$
\begin{aligned}
&= \bigcup_r [\![\tau]\!](L_1^r, \dots, L_n^r) \\
&\subseteq \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}} \alpha\}.
\end{aligned}
$$

□

Лема 2.2. Нека $L_1, \dots, L_n$ е някоя неподвижна точка на непрекъснатото изображение $\Gamma \stackrel{\text{деф}}{=} [\![\tau_1]\!] \times \dots \times [\![\tau_n]\!]$. За всеки терм τ с променливи измежду $X_1, \dots, X_n$ е изпълнено, че:

$$\{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}} \alpha\} \subseteq [\![\tau]\!](L_1, \dots, L_n).$$

Доказателство. Да означим с **Include**(ℓ) твърдението „за всеки терм τ с променливи измежду $X_1, \dots, X_n$ е изпълнено, че:

$$\{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}}^{\ell} \alpha\} \subseteq [\![\tau]\!](L_1, \dots, L_n)\text{''}.$$

Ще докажем, че $(\forall \ell \in \mathbb{N})$**Include**$(\ell)$.

Нека $\ell = 0$. Според правилата на операционната семантика, единственият случай за $\tau \Downarrow_{\mathtt{R}}^{0} \alpha$ е когато $\tau \equiv \varepsilon$. Тогава е ясно, че имаме **Include**(0), защото

$$\{\alpha \in \Sigma^\star \mid \varepsilon \Downarrow_{\mathtt{R}}^{0} \alpha\} = \{\varepsilon\} = [\![\varepsilon]\!](L_1, \dots, L_n).$$

Нека $\ell > 0$ и да приемем, че е изпълнено **Include**$(\ell - 1)$, което ще ни служи като индукционно предположение. Да разгледаме дума α и терм τ, за които $\tau \Downarrow_{\mathtt{R}}^{\ell} \alpha$. Според правилата на операционната семантика, имаме три случая в зависимост от това какво е последното правило, което сме приложили.

- Първият случай е когато имаме, че $\tau \equiv a \cdot X_j$ и

  $$\frac{\tau_j \Downarrow_{\mathtt{R}}^{\ell-1} \beta}{a \cdot X_j \Downarrow_{\mathtt{R}}^{\ell} a \cdot \beta} \text{ (rec)}$$

  Според правилата на операционната семантика имаме, че:

  $$\{a\} \cdot \{\beta \in \Sigma^\star \mid \tau_j \Downarrow_{\mathtt{R}}^{\ell-1} \beta\} = \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}}^{\ell} \alpha\} \tag{2.4}$$

  От дефиницията на денотационната семантика знаем, че:

  $$
  \begin{aligned}
  [\![\tau]\!](L_1, \dots, L_n) &= [\![a \cdot X_j]\!](L_1, \dots, L_n) \\
  &= \{a\} \cdot L_j \\
  &= \{a\} \cdot [\![\tau_j]\!](L_1, \dots, L_n)
  \end{aligned}
  $$

защото $L_1, \dots, L_n$ е *неподвижна точка* на изображението Γ и $L_j = [\![\tau_j]\!](L_1, \dots, L_n)$. От **Include**$(\ell - 1)$ имаме следното:

$$\{\beta \in \Sigma^\star \mid \tau_j \Downarrow_{\mathtt{R}}^{\ell-1} \beta\} \subseteq [\![\tau_j]\!](L_1, \dots, L_n). \tag{2.5}$$

Заключаваме, че:

$$\begin{aligned}
\{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{R}}^{\ell} \alpha\} &= \{a\} \cdot \{\beta \in \Sigma^\star \mid \tau_j \Downarrow_{\mathtt{R}}^{\ell-1} \beta\} && \text{// от Свойство (2.4)} \\
&\subseteq \{a\} \cdot [\![\tau_j]\!](L_1, \dots, L_n) && \text{// от Свойство (2.5)} \\
&= \{a\} \cdot L_j && \text{// } L_j = [\![\tau_j]\!](L_1, \dots, L_n) \\
&= [\![\tau]\!](L_1, \dots, L_n) && \text{// деф. на денот. сем.}
\end{aligned}$$

- Сега нека $\tau \equiv \rho + \mu$ и последното правило, което сме приложили е следното:

$$\frac{\rho \Downarrow_{\mathtt{R}}^{\ell-1} \alpha}{\rho + \mu \Downarrow_{\mathtt{R}}^{\ell} \alpha} \text{(left-or)}$$

Знаем, че $[\![\tau]\!](L_1, \dots, L_n) = [\![\rho]\!](L_1, \dots, L_n) \cup [\![\mu]\!](L_1, \dots, L_n)$. Понеже сме приели, че **Include**$(\ell - 1)$ е изпълнено, получаваме:

$$\begin{aligned}
\{\alpha \in \Sigma^\star \mid \rho \Downarrow_{\mathtt{R}}^{\ell-1} \alpha\} &\subseteq [\![\rho]\!](L_1, \dots, L_n) \\
&\subseteq [\![\tau]\!](L_1, \dots, L_n).
\end{aligned}$$

- Сега нека $\tau \equiv \rho + \mu$ и последното правило, което сме приложили е следното:

$$\frac{\mu \Downarrow_{\mathtt{R}}^{\ell-1} \alpha}{\rho + \mu \Downarrow_{\mathtt{R}}^{\ell} \alpha} \text{(right-or)}$$

Този случай е аналогичен на предния.

□

Така доказахме теоремата за еквивалентност на денотационната и операционната семантика.

Теорема 2.1 (Теорема за еквивалентност)**.** Нека $L_1, \dots, L_n$ е най-малката неподвижна точка на Γ. Тогава за всеки индекс i измежду $1, \dots, n$ е изпълнено, че:

$$L_i = \{\alpha \in \Sigma^\star \mid \tau_i \Downarrow_{\mathtt{R}} \alpha\}.$$

2.5 Разширения

2.5.1 Езикът REG++

Нека тук с α, β, γ да означаваме произволни регулярни изрази. Да въведем означението $\alpha \leq \beta$, ако $\mathcal{L}(\alpha) \subseteq \mathcal{L}(\beta)$.

Задача 2.5. Докажете, че ако $\alpha \cdot \gamma + \beta \leq \gamma$, то $\alpha^\star\beta \leq \gamma$.

Упътване. Докажете, че $\alpha^n \cdot \beta \leq \gamma$ за всяко естествено число n. □

Да разгледаме следното разширение на езика REG , което да наречем REG ++. Термовете на езика REG++ се формират спрямо следната абстрактна граматика:

Новото е, че тук позволяваме да имаме термове от вида $\alpha \cdot X$, за произволен регулярен израз α, докато в езика REG позволявахме само $a \cdot X$, за произволна буква a.

$$\tau ::= \alpha \mid \alpha \cdot X \mid \tau_1 + \tau_2,$$

където с α означаваме произволен регулярен израз над предварително фиксирана азбука Σ, т.е.

$$\alpha ::= \emptyset \mid \varepsilon \mid a \mid \alpha_1 + \alpha_2 \mid \alpha_1 \cdot \alpha_2 \mid \alpha_1^\star,$$

където $a \in \Sigma$.

Задача 2.6 (Правило на Ардън [2]). Да разгледаме програмата на езика REG++ съставена само от реда:

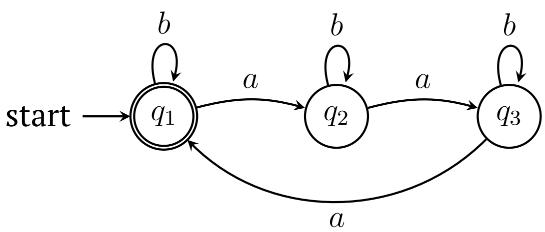
$$X = \alpha \cdot X + \beta.$$

Докажете, че най-малкото решение на уравнението, което съотвества на тази програма е езикът описан с регулярния израз $\alpha^\star\beta$. Докажете, че ако $\varepsilon \notin \mathcal{L}(\alpha)$, то това решение е и единствено.

Правилото на Ардън ни дава друг метод, с чиято помощ можем да намерим най-малкото решение на система от уравнения на езика REG . Ще илюстрираме този метод с един пример, който вече решихме.

Пример 2.5. Да разгледаме следния краен детерминиран автомат:

Означаваме с $|\omega|_a$ броя на срещанията на a в думата ω.



Фигура 2.5: Автомат разпознаващ $\{\omega \in \{a,b\}^\star \mid |\omega|_a \equiv 0 \mod 3\}$

На този автомат съответства следната програма R, която е програма на езика REG++ (а също и на езика REG):

Тук $\Sigma = \{a, b\}$.

$$
\left|\begin{array}{lcl}
X_1 & = & a \cdot X_2 + b \cdot X_1 + \varepsilon \\
X_2 & = & a \cdot X_3 + b \cdot X_2 \\
X_3 & = & a \cdot X_1 + b \cdot X_3.
\end{array}\right.
$$

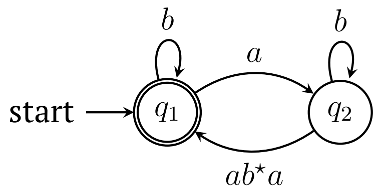
Да видим как можем да намерим най-малкото решение на тази система като използваме само правилото на Ардън. Нека първо да разгледаме само последния ред на системата като си мислим, че имаме азбука $\Sigma_3 = \{a, b, X_1, X_2\}$, т.е. единствената свободна променлива е X_3. Тогава получаваме следното:

$$X_3 = \underbrace{b}_{\alpha} \cdot X_3 + \underbrace{a \cdot X_1}_{\beta}.$$

Според правилото на Ардън, най-малкото решение на тази система е езикът описван с регулярния израз $b^\star a \cdot X_1$ над азбуката Σ_3. Заместваме X_3 с този израз в горните две уравнения и получаваме нова програма на езика REG++ :

$$
\left|\begin{array}{lcl}
X_1 & = & b \cdot X_1 + a \cdot X_2 + \varepsilon \\
X_2 & = & b \cdot X_2 + ab^\star a \cdot X_1.
\end{array}\right.
$$

Ако позволим да имаме регулярни изрази по ребрата на автомат, то новият автомат би бил този:



Сега разглеждаме втория ред като си мислим, че имаме азбуката $\Sigma_2 = \{a, b, X_1\}$, т.е. едиствената свободна променлива е X_2. Тогава получаваме следното:

$$X_2 = \underbrace{b}_{\alpha} \cdot X_2 + \underbrace{ab^\star a \cdot X_1}_{\beta}.$$

Според правилото на Ардън, най-малкото решение на тази система е езикът описван с регулярния израз $b^\star ab^\star a \cdot X_1$ над азбуката Σ_2. Заместваме X_2 с израза $b^\star ab^\star a \cdot X_1$ в първия ред и получаваме:

$$X_1 = b \cdot X_1 + ab^\star ab^\star a \cdot X_1 + \varepsilon.$$

Оттук имаме, че:

$$X_1 = \underbrace{(b + ab^\star ab^\star a)}_{\alpha} \cdot X_1 + \underbrace{\varepsilon}_{\beta}.$$

Така получаваме, че най-малкото решение на първия ред е $(b + ab^\star ab^\star a)^\star$. Заключаваме, че най-малкото решение на системата е тройката от езици:

$$
\left|\begin{array}{lcl}
\hat{L}_1 & = & (b + ab^\star ab^\star a)^\star \\
\hat{L}_2 & = & b^\star ab^\star a(b + ab^\star ab^\star a)^\star \\
\hat{L}_3 & = & b^\star a(b + ab^\star ab^\star a)^\star.
\end{array}\right.
$$

Тук един от проблемите е, че изобщо не е ясно дали тези регулярни изрази са „оптимални”, т.е. може да съществуват по-кратки регулярни изрази, които да описват същите езици.

Да напомним, че вече намерихме следното представяне на най-малкото решение на системата:

$$\begin{aligned}
\hat{L}_1 &= \{\omega \in \{a,b\}^\star \mid |\omega|_a \equiv 0 \bmod 3\}\\
\hat{L}_2 &= \{\omega \in \{a,b\}^\star \mid |\omega|_a \equiv 2 \bmod 3\}\\
\hat{L}_3 &= \{\omega \in \{a,b\}^\star \mid |\omega|_a \equiv 1 \bmod 3\}.
\end{aligned}$$

2.5.2 Езикът CFG

[7, стр. 121]

Термовете на езика CFG се описват със следната абстрактна граматика:

$$\tau ::= \varepsilon \mid a \mid X \mid \tau \cdot \tau \mid \tau + \tau.$$

Нека оттук нататък да приемем, че сме фиксирали една програма G на езика CFG :

$$\left|\begin{array}{lcl}
X_1 & = & \tau_1[X_1, \dots, X_n]\\
 & \vdots & \\
X_n & = & \tau_n[X_1, \dots, X_n].
\end{array}\right.$$

Операционна семантика

С индукция по ℓ, дефинираме релацията $\tau \Downarrow_{\mathtt{G}}^{\ell} \alpha$, където τ е терм на езика CFG със свободни променливи измежду $X_1, \dots, X_n$.

$$\text{(eps)}\ \frac{}{\varepsilon \Downarrow_{\mathtt{G}}^{0} \varepsilon} \qquad \frac{}{a \Downarrow_{\mathtt{G}}^{1} a}\ \text{(letter)}$$

$$\text{(left-or)}\ \frac{\rho \Downarrow_{\mathtt{G}}^{\ell} \alpha}{\rho + \mu \Downarrow_{\mathtt{G}}^{\ell+1} \alpha} \qquad \frac{\rho \Downarrow_{\mathtt{G}}^{\ell} \alpha}{\rho + \mu \Downarrow_{\mathtt{G}}^{\ell+1} \alpha}\ \text{(right-or)}$$

$$\text{(rec)}\ \frac{\tau_i \Downarrow_{\mathtt{G}}^{\ell} \alpha}{X_i \Downarrow_{\mathtt{G}}^{\ell+1} \alpha} \qquad \frac{\rho \Downarrow_{\mathtt{G}}^{\ell_1} \alpha \quad \mu \Downarrow_{\mathtt{G}}^{\ell_2} \beta}{\rho \cdot \mu \Downarrow_{\mathtt{G}}^{\ell_1+\ell_2+1} \alpha \cdot \beta}\ \text{(concat)}$$

Фигура 2.6: Правила на операционната семантика на езика CFG .

Ще пишем $\tau \Downarrow_{\mathtt{G}} \alpha$ точно тогава, когато съществува ℓ, за което $\tau \Downarrow_{\mathtt{G}}^{\ell} \alpha$.

Денотационна семантика

За всеки терм $\tau[X_1, \dots, X_n]$ на езика CFG дефинираме изображението

$$[\![\tau]\!] : \mathcal{P}(\Sigma^\star)^n \to \mathcal{P}(\Sigma^\star)$$

по следния начин:

- $[\![\varepsilon]\!](L_1, \ldots, L_n) = \{\varepsilon\}$.
- $[\![a]\!](L_1, \ldots, L_n) = \{a\}$.
- $[\![X_j]\!](L_1, \ldots, L_n) = L_j$.
- $[\![\sigma + \rho]\!](L_1, \ldots, L_n) = [\![\sigma]\!](L_1, \ldots, L_n) \cup [\![\rho]\!](L_1, \ldots, L_n)$.
- $[\![\sigma \cdot \rho]\!](L_1, \ldots, L_n) = [\![\sigma]\!](L_1, \ldots, L_n) \cdot [\![\rho]\!](L_1, \ldots, L_n)$.

Задача 2.7. Докажете, че за всеки терм τ на езика REG имаме, че $[\![\tau]\!]$ е непрекъснато изображение в областта на Скот $\mathcal{S} = (\mathcal{P}(\Sigma^\star), \subseteq, \emptyset)$.

Тук правилото на Ардън не работи!

Пример 2.6. Да разгледаме следната програма на езика CFG .

$$\begin{aligned} X &= \overbrace{a \cdot X \cdot c + Y}^{\tau_1[X,Y]} \\ Y &= \underbrace{b \cdot Y \cdot c + \varepsilon}_{\tau_2[X,Y]}. \end{aligned}$$

На тази програма съответства системата от непрекъснати изображения:

$$\begin{aligned} L &= [\![\tau_1]\!](L, M) = \{a\} \cdot L_1 \cdot \{c\} \cup L_2 \\ M &= [\![\tau_2]\!](L_1, L_2) = \{b\} \cdot L_2 \cdot \{c\} \cup \{\varepsilon\}. \end{aligned}$$

Да намерим най-малкото решение $(\hat{L}, \hat{M})$ на тази система.

$$\begin{aligned} L_0 &= \emptyset \\ M_0 &= \emptyset \\ L_1 &= [\![\tau_1]\!](L_0, M_0) \\ &= \{a\} \cdot \emptyset \cdot \{c\} \cup \emptyset = \emptyset \\ M_1 &= [\![\tau_2]\!](L_0, M_0) \\ &= \{b\} \cdot \emptyset \cdot \{c\} \cup \{\varepsilon\} = \{\varepsilon\} \\ L_2 &= \{a\} \cdot L_1 \cdot \{c\} \cup M_1 = \{\varepsilon\} \\ M_2 &= \{b\} \cdot M_1 \cdot \{c\} \cup M_1 \\ &= \{b\} \cdot \{\varepsilon\} \cdot \{c\} \cup \{\varepsilon\} = \{bc, \varepsilon\} \\ L_3 &= \{a\} \cdot L_2 \cdot \{c\} \cup M_2 \\ &= \{ac, bc, \varepsilon\} \\ &= \{a^m b^n c^k \mid 2 > m + n = k\} \\ M_3 &= \{b\} \cdot M_2 \cdot \{c\} \cup M_2 \\ &= \{b\} \cdot \{bc, \varepsilon\} \cdot \{c\} \cup \{bc, \varepsilon\} \\ &= \{b^n c^n \mid 3 > n\}. \end{aligned}$$

✍ Довършете доказателството!

Сега вече сме готови да формулираме нашето индукционно предположение:

$$L_t = \{a^m b^n c^k \mid t - 1 > m + n = k\}$$

$$M_t = \{b^n c^n \mid t > n\}.$$

Докажете, че $\hat{L} = \{a^m b^n c^k \mid m + n = k\}$ и $\hat{M} = \{b^n c^n \mid n \in \mathbb{N}\}$.

Задача 2.8 (Теорема за характеризация)**.** Да разгледаме системата от непрекъснати оператори

$$\left|\begin{array}{lcl} L_1 & = & [\![\tau_1]\!](L_1, \dots, L_n) \\ & \vdots & \\ L_n & = & [\![\tau_n]\!](L_1, \dots, L_n). \end{array}\right.$$

Знаем, че тя притежава най-малко решение $(\hat{L}_1, \dots, \hat{L}_n)$. Докажете, че всеки от езиците $\hat{L}_i$ е безконтекстен.

Докажете, че всеки безконтекстен език е елемент от най-малкото решение на някоя система от непрекъснати изображения от горния вид.

Еквивалентност

Задача 2.9. Докажете, че за всеки индекс r и всеки терм τ на езика CFG с променливи измежду $X_1, \dots, X_n$ е изпълнено, че:

$$[\![\tau]\!](L_1^r, \dots, L_n^r) \subseteq \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{G}} \alpha\}.$$

Заключете, че ако $L_1, \dots, L_n$ е най-малкото решение на непрекъснатото изображение $\Gamma \stackrel{\text{деф}}{=} [\![\tau_1]\!] \times \cdots \times [\![\tau_n]\!]$, то за произволен терм τ с променливи измежду $X_1, \dots, X_n$ е изпълнено, че:

$$[\![\tau]\!](L_1, \dots, L_n) \subseteq \{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{G}} \alpha\}.$$

Задача 2.10. Нека $L_1, \dots, L_n$ е неподвижна точка на $\Gamma \stackrel{\text{деф}}{=} [\![\tau_1]\!] \times \cdots \times [\![\tau_n]\!]$. Докажете, че за всеки терм τ на езика CFG с променливи измежду $X_1, \dots, X_n$ е изпълнено, че:

$$\{\alpha \in \Sigma^\star \mid \tau \Downarrow_{\mathtt{G}} \alpha\} \subseteq [\![\tau]\!](L_1, \dots, L_n).$$

Заключете, че ако $L_1, \dots, L_n$ е най-малката неподвижна точка на Γ, то за всеки индекс i измежду $1, \dots, n$ е изпълнено, че:

$$L_i = \{\alpha \in \Sigma^\star \mid \tau_i \Downarrow_{\mathtt{G}} \alpha\}.$$

Глава 3

Езикът FUN

3.1 Синтаксис

В тази глава до голяма степен следваме [12, Глава 9]. Там този език се нарича REC.

Ще разгледаме един много прост език за функционално програмиране.

За разлика от [14], ще подходим както в [12] и няма да въвеждаме термове от тип Bool.

- константи $\mathtt{n}$, за всяко число $n \in \mathbb{N}$;

Константите не са числа! Константите са синтактични обекти, докато числата са семантични обекти. Обърнете внимание, че в езика имаме константи за всяко естествено число, но в езика нямаме константа за $\bot$. Това е една разлика с хаскел, където в езика има константа за $\bot$ и тя е означена с `undefined`.

- изброимо много променливи от тип 0 (или обектови променливи) $\mathtt{x}, \mathtt{y}, \mathtt{z}, \dots$, евентуално с индекси;
- изброимо много променливи от тип 1 (или функционални променливи) $\mathtt{f}, \mathtt{g}, \mathtt{h}, \dots$, евентуално с индекси. Формално погледнато, трябва на всяка функционална променлива $\mathtt{f}$ да съпоставим число - брой аргументи, които приема. Нека да означим с $\sharp\mathtt{f}$ броя аргументи на $\mathtt{f}$. Обикновено броят аргументи на $\mathtt{f}$ ще е ясен от контекста.

Удобно е в нашия език още на синтактично ниво да правим разлика между двата типа променливи, които имаме в езика.

- Термовете, които обикновено ще означаваме с τ, в езика FUN те се формират според следната абстрактна граматика:

$$\tau ::= \mathtt{n} \mid \mathtt{x} \mid \tau_1 + \tau_2 \mid \tau_1 - \tau_2 \mid \tau_1 \,\mathtt{==}\, \tau_2 \mid \mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3 \mid \mathtt{f}(\tau_1, \dots, \tau_m).$$

Тук $m = \sharp\mathtt{f}$

Граматиката е във форма на Бекус-Наур.

- Ще записваме $\tau[\mathtt{f}_1, \dots, \mathtt{f}_k, \mathtt{x}_1, \dots, \mathtt{x}_n]$, когато искаме да означим, че променливите на терма τ са *измежду* посочените.
- Ще наричаме един терм **функционален**, ако той не съдържа обектови променливи. Обикновено ще означаваме функционалните термове с μ, а произволни термове с τ.
- Най-удобно е да си мислим за един терм като за дърво.

[12, стр. 13].

- Ще пишем $\tau_1 \equiv \tau_2$, ако термовете τ_1 и τ_2 представляват едни и същи дървета. Например, $2 + 3 = 3 + 2$, но $2 + 3 \not\equiv 3 + 2$.
- С $\tau[\mathtt{x}/\mu]$ ще означаваме терма получен от τ, в който всяко срещане на обектовата променлива $\mathtt{x}$ е заменена с функционалния терм μ. Можем да дадем формална дефиниция с индукция по построението на термовете:

Тук нямаме усложнения с дефиницията на замяната, защото нямаме свързани променливи.

- Ако $\tau \equiv \mathtt{n}$, то
$$\mathtt{n}[\mathtt{x}/\mu] \equiv \mathtt{n}.$$
- Ако $\tau \equiv \mathtt{x}$, то
$$\mathtt{x}[\mathtt{x}/\mu] \equiv \mu.$$
- Ако $\tau \equiv \mathtt{y}$ и $\mathtt{y} \not\equiv \mathtt{x}$, то
$$\mathtt{y}[\mathtt{x}/\mu] \equiv \mathtt{y}.$$
- Ако $\tau \equiv \tau_1 + \tau_2$, то
$$\tau[x/\mu] \equiv \tau_1[x/\mu] + \tau_2[x/\mu].$$
- Ако $\tau \equiv \tau_1 - \tau_2$, то
$$\tau[x/\mu] \equiv \tau_1[x/\mu] - \tau_2[x/\mu].$$
- Ако $\tau \equiv \tau_1$ == τ_2, то
$$\tau[x/\mu] \equiv \tau_1[x/\mu] \texttt{ == } \tau_2[x/\mu].$$
- Ако $\tau \equiv \texttt{if } \tau_1 \texttt{ then } \tau_2 \texttt{ else } \tau_3$, то
$$\tau[x/\mu] \equiv \texttt{if } \tau_1[x/\mu] \texttt{ then } \tau_2[x/\mu] \texttt{ else } \tau_3[x/\mu].$$
- Ако $\tau \equiv \mathtt{f}(\tau_1, \dots, \tau_m)$, то
$$\tau[x/\mu] \equiv \mathtt{f}(\tau_1[x/\mu], \dots, \tau_m[x/\mu]).$$

Пример 3.1. Пример как се прилага замяната.

Една **рекурсивна програма** P на езика FUN има следния общ вид:

$$\mathtt{P} = \begin{cases} \mathtt{f}_1(\mathtt{x}_1, \dots, \mathtt{x}_{m_1}) = \tau_1[\mathtt{f}_1, \dots, \mathtt{f}_k, \mathtt{x}_1, \dots, \mathtt{x}_{m_1}] \\ \vdots \\ \mathtt{f}_k(\mathtt{x}_1, \dots, \mathtt{x}_{m_k}) = \tau_k[\mathtt{f}_1, \dots, \mathtt{f}_k, \mathtt{x}_1, \dots, \mathtt{x}_{m_k}] \end{cases}$$

[12, стр. 141]

Една програма е просто текст със специален формат. Важният въпрос е каква функция (семантика) отговаря на този текст (синтаксис)

Може да си мислите, че $\mathtt{f}_1$ е `main` функцията на нашата програма. Тук важното е, че в термовете $\tau_1, \dots, \tau_k$, които дефинират $\mathtt{f}_1, \dots, \mathtt{f}_k$ могат да участват функционални променливи само измежду $\mathtt{f}_1, \dots, \mathtt{f}_k$.

В такъв случай казваме, че термът τ_i задава *дефиницията* на фунционалната променлива $\mathtt{f}_i$.

Пример 3.2. Да разгледаме програмата P на езика FUN :

```
h(x) = f(x, 1)
f(x, y) = if x == y then 0
             else f(x, y+1) + 1
```

Да положим

$$\tau_1[\mathtt{h}, \mathtt{f}, \mathtt{x}] \overset{\text{деф}}{\equiv} \mathtt{f}(\mathtt{x}, 1)$$
$$\tau_2[\mathtt{h}, \mathtt{f}, \mathtt{x}, \mathtt{y}] \overset{\text{деф}}{\equiv} \texttt{if x == y then 0 else f(x,y+1) + 1}.$$

Тогава програмата P приема следния вид:

$$\mathtt{h}(\mathtt{x}) = \tau_1[\mathtt{h}, \mathtt{f}, \mathtt{x}]$$
$$\mathtt{f}(\mathtt{x}, \mathtt{y}) = \tau_2[\mathtt{h}, \mathtt{f}, \mathtt{x}, \mathtt{y}].$$

3.2 Денотационна семантика

3.2.1 Термални оператори

[12, стр. 155]

За всеки терм $\tau[\mathtt{f}_1, \dots, \mathtt{f}_k, \mathtt{x}_1, \dots, \mathtt{x}_n]$, ще разгледаме изображението със сигнатура

$$[\![\tau]\!] : [\mathbb{N}_\perp^{m_1} \xrightarrow{\text{н}} \mathbb{N}_\perp] \times \cdots \times [\mathbb{N}_\perp^{m_k} \xrightarrow{\text{н}} \mathbb{N}_\perp] \to [\mathbb{N}_\perp^n \to \mathbb{N}_\perp],$$

което ще дефинираме с индукция по построението на термовете. Изображенията от вида $[\![\tau]\!]$ ще наричаме **термални оператори**.

- ако $\tau \equiv \mathtt{c}$, за някоя константа $\mathtt{c}$, то

$$[\![\mathtt{c}]\!](\overline{\varphi})(\overline{a}) \stackrel{\text{деф}}{=} c.$$

- ако $\tau \equiv \mathtt{x}_i$, за някоя обектова променлива $\mathtt{x}_i$, то

$$[\![\mathtt{x}_i]\!](\overline{\varphi})(\overline{a}) \stackrel{\text{деф}}{=} a_i.$$

- ако $\tau \equiv \tau_1 + \tau_2$, то

$$[\![\tau_1 + \tau_2]\!](\overline{\varphi})(\overline{a}) \stackrel{\text{деф}}{=} (\mathtt{+})([\![\tau_1]\!](\overline{\varphi})(\overline{a}), [\![\tau_2]\!](\overline{\varphi})(\overline{a})).$$

- ако $\tau \equiv \tau_1 - \tau_2$, то

$$[\![\tau_1 - \tau_2]\!](\overline{\varphi})(\overline{a}) \stackrel{\text{деф}}{=} (\mathtt{-})([\![\tau_1]\!](\overline{\varphi})(\overline{a}), [\![\tau_2]\!](\overline{\varphi})(\overline{a})).$$

От *Задача* 1.6 знаем, че изображенията (+), (-), (==) са непрекъснати.

- ако $\tau \equiv \tau_1 \mathtt{==} \tau_2$, то

$$[\![\tau_1 \mathtt{==} \tau_2]\!](\overline{\varphi})(\overline{a}) \stackrel{\text{деф}}{=} (\mathtt{==})([\![\tau_1]\!](\overline{\varphi})(\overline{a}), [\![\tau_2]\!](\overline{\varphi})(\overline{a})).$$

От *Задача* 1.9 знаем, че if е непрекъснато изображение.

- ако $\tau \equiv$ if τ_1 then τ_2 else τ_3, то

$$[\![\mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3]\!](\overline{\varphi})(\overline{a}) \stackrel{\text{деф}}{=} \mathtt{if}_{\mathbb{N}_\perp}([\![\tau_1]\!](\overline{\varphi})(\overline{a}), [\![\tau_2]\!](\overline{\varphi})(\overline{a}), [\![\tau_3]\!](\overline{\varphi})(\overline{a})).$$

Термовете τ_j може да имат различен брой променливи. Ако се наложи, разширяваме ги с фиктивни променливи

- ако $\tau \equiv \mathtt{f}_i(\tau_1, \dots, \tau_{m_i})$, то

$$[\![\mathtt{f}_i(\tau_1, \dots, \tau_{m_i})]\!](\overline{\varphi})(\overline{a}) \stackrel{\text{деф}}{=} \varphi_i([\![\tau_1]\!](\overline{\varphi})(\overline{a}), \dots, [\![\tau_{m_i}]\!](\overline{\varphi})(\overline{a})).$$

Лема 3.1 (Лема за замяната)**.** Да разгледаме терма $\tau[\mathtt{f}_1, \dots, \mathtt{f}_k, \mathtt{x}_1, \dots, \mathtt{x}_n]$ и *функционалните* термове $\mu_1[\mathtt{f}_1, \dots, \mathtt{f}_k], \dots, \mu_n[\mathtt{f}_1, \dots, \mathtt{f}_k]$. Тогава

$$[\![\tau[\overline{\mathtt{x}}/\overline{\mu}]]\!](\overline{\varphi}) = [\![\tau]\!](\overline{\varphi})([\![\mu_1]\!](\overline{\varphi}), \dots, [\![\mu_n]\!](\overline{\varphi}))$$

за произволни $\varphi_i \in [\mathbb{N}_\perp^{m_i} \xrightarrow{\text{н}} \mathbb{N}_\perp]$, за $i = 1, \dots, k$.

Доказателство. Доказателството се провежда с *индукция по построението на терма* τ.

Понеже нямаме обектови променливи в $\mu_1, \dots, \mu_n$, то е очевидно как правим замяната. Тук следваме [12, стр. 149]. Това твърдение е без доказателство в [14, стр. 188].

- Нека да започнем с най-лесния случай. Нека $\tau \equiv \mathtt{c}$, за някоя константа. Тогава $\tau[\overline{\mathtt{x}}/\overline{\mu}] \equiv \mathtt{c}$. Това означава, че

$$[\![\tau[\overline{\mathtt{x}}/\overline{\mu}]]\!](\overline{\varphi}) = [\![\mathtt{c}]\!](\overline{\varphi}) \stackrel{\text{деф}}{=} c.$$

c е константа, докато $c \in \mathbb{N}_\perp$

От друга страна, имаме също, че

$$[\![\mathtt{c}]\!](\overline{\varphi})([\![\mu_1]\!](\overline{\varphi}), \dots, [\![\mu_n]\!](\overline{\varphi})) \stackrel{\text{деф}}{=} c.$$

- Нека $\tau \equiv \mathtt{x}_i$. Тогава $\mathtt{x}_i[\bar{\mathtt{x}}/\bar{\mu}] \equiv \mu_i$ и следователно

$$[\![\mathtt{x}_i[\bar{\mathtt{x}}/\bar{\mu}]]\!](\bar{\varphi}) = [\![\mu_i]\!](\bar{\varphi}).$$

От друга страна, по дефиниция на стойност на терм,

$$[\![\mathtt{x}_i]\!](\overline{\varphi})([\![\mu_1]\!](\bar{\varphi}), \dots, [\![\mu_n]\!](\bar{\varphi})) = [\![\mu_i]\!](\bar{\varphi}).$$

- Нека $\tau \equiv \tau_1 + \tau_2$. От **(И.П.)** имаме, че за $j = 1, 2$ е изпълнено следното:

$$[\![\tau_j[\overline{\mathtt{x}}/\overline{\mu}]]\!](\overline{\varphi}) = [\![\tau_j]\!](\overline{\varphi})(\underbrace{[\![\mu_1]\!](\overline{\varphi})}_{b_1}, \dots, \underbrace{[\![\mu_n]\!](\overline{\varphi})}_{b_n}).$$

Тогава

$$\begin{aligned}
[\![\tau[\overline{\mathtt{x}}/\overline{\mu}]]\!](\overline{\varphi}) &= [\![\tau_1[\overline{\mathtt{x}}/\overline{\mu}] + \tau_2[\overline{\mathtt{x}}/\overline{\mu}]]\!](\overline{\varphi}) \\
&= (\mathtt{+})([\![\tau_1[\overline{\mathtt{x}}/\overline{\mu}]]\!](\overline{\varphi}), [\![\tau_2[\overline{\mathtt{x}}/\overline{\mu}]]\!](\overline{\varphi})) && \text{// от деф.} \\
&= (\mathtt{+})([\![\tau_1]\!](\overline{\varphi})(b_1, \dots, b_n), [\![\tau_2]\!](\overline{\varphi})(b_1, \dots, b_n)) && \text{// от } \textbf{(И.П.)} \\
&= [\![\tau]\!](\overline{\varphi})(b_1, \dots, b_n) && \text{// от деф.} \\
&= [\![\tau]\!](\overline{\varphi})([\![\mu_1]\!](\bar{\varphi}), \dots, [\![\mu_n]\!](\overline{\varphi})). && \text{// } b_j \stackrel{\text{деф}}{=} [\![\mu_j]\!](\overline{\varphi})
\end{aligned}$$

✍ Докажете сами останалите случаи. Те не крият изненади.

- Нека $\tau \equiv \tau_1 \ \mathtt{-} \ \tau_2$.
- Нека $\tau \equiv \tau_1 \mathtt{==} \tau_2$.
- Нека $\tau \equiv \mathtt{if}\ \tau_0\ \mathtt{then}\ \tau_1\ \mathtt{else}\ \tau_2$.
- Нека $\tau \equiv \mathtt{f}_i(\tau_1, \dots, \tau_{m_i})$. Имаме, че

$$\tau[\bar{\mathtt{x}}/\bar{\mu}] \equiv \mathtt{f}_i(\tau_1[\bar{\mathtt{x}}/\bar{\mu}], \dots, \tau_{m_i}[\bar{\mathtt{x}}/\bar{\mu}]). \tag{3.1}$$

Нека за улеснение да означим $b_i \stackrel{\text{деф}}{=} [\![\mu_i]\!](\bar{\varphi})$, за $i = 1, \dots, n$. Прилагаме **(И.П.)** за термовете $\tau_1, \dots, \tau_{m_i}$ и получаваме за $j = 1, \dots, m_i$,

$$[\![\tau_j[\bar{\mathtt{x}}/\bar{\mu}]]\!](\bar{\varphi}) = [\![\tau_j]\!](\overline{\varphi})(\underbrace{[\![\mu_1]\!](\bar{\varphi})}_{b_1}, \dots, \underbrace{[\![\mu_n]\!](\bar{\varphi})}_{b_n}) \qquad \text{// от } \textbf{(И.П.)}$$

$$= [\![\tau_j]\!](\overline{\varphi})(b_1, \ldots, b_n) \qquad\qquad // \ b_i \stackrel{\text{деф}}{=} [\![\mu_i]\!](\bar{\varphi}).$$

Следователно,

$$\begin{aligned}
[\![\tau[\bar{\mathtt{x}}/\overline{\mu}]]\!](\overline{\varphi}) &= [\![\mathtt{f}_i(\tau_1[\bar{\mathtt{x}}/\overline{\mu}], \ldots, \tau_{m_i}[\bar{\mathtt{x}}/\overline{\mu}])]\!](\overline{\varphi}) && // \text{ от } (3.1)\\
&= \varphi_i([\![\tau_1[\bar{\mathtt{x}}/\overline{\mu}]]\!](\overline{\varphi}), \ldots, [\![\tau_{m_i}[\bar{\mathtt{x}}/\overline{\mu}]]\!](\overline{\varphi})) && // \text{ от деф.}\\
&= \varphi_i([\![\tau_1]\!](\overline{\varphi})(\bar{b}), \ldots, [\![\tau_{m_i}]\!](\overline{\varphi})(\bar{b})) && // \text{ от } \textbf{(И.П.)}\\
&= [\![\mathtt{f}_i(\tau_1, \ldots, \tau_{m_i})]\!](\overline{\varphi})(\bar{b}) && // \text{ от деф.}\\
&= [\![\tau]\!](\overline{\varphi})(\bar{b}) && // \ \tau \equiv \mathtt{f}_i(\tau_1, \ldots, \tau_{m_i})\\
&= [\![\tau]\!](\overline{\varphi})([\![\mu_1]\!](\overline{\varphi}), \ldots, [\![\mu_n]\!](\overline{\varphi})) && // \ b_i \stackrel{\text{деф}}{=} [\![\mu_i]\!](\overline{\varphi}).
\end{aligned}$$

□

3.2.2 Непрекъснатост на термалните оператори

Спокойно, след като докажем следващите две леми, няма да има нужда никога повече да правим такива сложни преобразувания.

Пример 3.3. Да разгледаме терма $\tau[f_1, f_2, x] \equiv f_1(x) + f_2(x)$. Да видим какво представлява $[\![\tau]\!]$.

$$\begin{aligned}
[\![f_1(x) + f_2(x)]\!](\varphi, \psi)(a) &= \mathtt{(+)}([\![f_1(x)]\!](\varphi, \psi)(a), [\![f_2(x)]\!](\varphi, \psi)(a))\\
&= \mathtt{(+)}(\varphi(a), \psi(a))\\
&= (\mathtt{(+)} \circ (\varphi \times \psi))(a).
\end{aligned}$$

Или казано по-накратко,

$$[\![\tau]\!](\varphi, \psi) = \mathtt{(+)} \circ (\varphi \times \psi).$$

Оттук веднага се вижда, че ако φ и ψ са непрекъснати, то и $[\![\tau]\!](\varphi, \psi)$ също ще е непрекъснато изображение, т.е. $[\![\tau]\!](\varphi, \psi) \in [\mathbb{N}_\perp \stackrel{\text{н}}{\to} \mathbb{N}_\perp]$.

Можем да видим, че самото изображение $[\![\tau]\!]$ също е непрекъснато, т.е.

$$[\![\tau]\!] \in [[\mathbb{N}_\perp \stackrel{\text{н}}{\to} \mathbb{N}_\perp] \times [\mathbb{N}_\perp \stackrel{\text{н}}{\to} \mathbb{N}_\perp] \stackrel{\text{н}}{\to} [\mathbb{N}_\perp \stackrel{\text{н}}{\to} \mathbb{N}_\perp]].$$

За да направим това е достатъчно да си припомним основните непрекъснати изображения, които разгледахме в първа глава и да съобразим следното:

$$[\![\tau]\!](\varphi, \psi) = \mathtt{(.)}(\mathtt{(+)}, \mathtt{cross}(\varphi, \psi)).$$

Тогава за произволни вериги $(\varphi_i)_{i=0}^\infty$ и $(\psi_i)_{i=0}^\infty$ е изпълнено, че:

$$\begin{aligned}
[\![\tau]\!](\bigsqcup_i \varphi_i, \bigsqcup_i \psi_i) &= \mathtt{(.)}(\mathtt{(+)}, \mathtt{cross}(\bigsqcup_i \varphi_i, \bigsqcup_i \psi_i))\\
&= \mathtt{(.)}(\mathtt{(+)}, \bigsqcup_i \mathtt{cross}(\varphi_i, \psi_i))\\
&= \bigsqcup_i \mathtt{(.)}(\mathtt{(+)}, \mathtt{cross}(\varphi_i, \psi_i))\\
&= \bigsqcup_i [\![\tau]\!](\varphi_i, \psi_i).
\end{aligned}$$

Сега ще видим, че разсъжденията, които направихме в горния пример могат да се обобщят за всеки терм.

В този раздел всички доказателства се провеждат с индукция по построението на термовете.

Твърдение 3.1. За всеки терм $\tau[\mathtt{f}_1,\dots,\mathtt{f}_k,\mathtt{x}_1,\dots,\mathtt{x}_n]$ и произволни $\varphi_i \in [\mathbb{N}_\perp^{m_i} \xrightarrow{\text{н}} \mathbb{N}_\perp]$, за $i = 1,\dots,k$, е изпълнено, че

$$[\![\tau]\!](\overline{\varphi}) \in [\mathbb{N}_\perp^n \xrightarrow{\text{н}} \mathbb{N}_\perp].$$

Озн. $\overline{a}_i = (a_i^1,\dots,a_i^n)$. Тук можеше да искаме и $\varphi_i \in [\mathbb{N}_\perp^{m_i} \xrightarrow{\text{м}} \mathbb{N}_\perp]$.

Упътване. Индукция по построението на терма τ.

- Нека $\tau \equiv \mathtt{a}$. Тогава
$$[\![\tau]\!](\overline{\varphi}) = \mathtt{const}_a^n.$$

$\mathtt{const}_a^n \in [\mathbb{N}_\perp^n \xrightarrow{\text{н}} \mathbb{N}_\perp]$.

- Нека $\tau \equiv \mathtt{x}_i$. Тогава
$$[\![\tau]\!](\overline{\varphi}) = \mathtt{proj}_i^{\mathbb{N}_\perp,n}.$$

$\mathtt{proj}_i^{\mathbb{N}_\perp,n} \in [\mathbb{N}_\perp^n \xrightarrow{\text{н}} \mathbb{N}_\perp]$

- Нека $\tau \equiv \tau_1 + \tau_2$. Можем лесно да съобразим, че
$$[\![\tau_1 \texttt{+} \tau_2]\!](\overline{\varphi}) = (\texttt{+}) \circ ([\![\tau_1]\!](\overline{\varphi}) \times [\![\tau_2]\!](\overline{\varphi})).$$

За произволно $\overline{a} \in \mathbb{N}_\perp^n$,

$[\![\tau_1 + \tau_2]\!](\overline{\varphi})(\overline{a}) =$
$(\texttt{+})([\![\tau_1]\!](\overline{\varphi})(\overline{a}), [\![\tau_2]\!](\overline{\varphi})(\overline{a})) =$
$(\texttt{+})(([\![\tau_1]\!](\overline{\varphi}) \times [\![\tau_2]\!](\overline{\varphi}))(\overline{a})) =$
$((\texttt{+}) \circ ([\![\tau_1]\!](\overline{\varphi}) \times [\![\tau_2]\!](\overline{\varphi})))(\overline{a})$.

  Първо, от **(И.П.)** имаме, че $[\![\tau_1]\!](\overline{\varphi})$ и $[\![\tau_2]\!](\overline{\varphi})$ са непрекъснати. От *Твърдение* 1.9 знаем, $[\![\tau_1]\!](\overline{\varphi}) \times [\![\tau_2]\!](\overline{\varphi})$ е непрекъснато изображение. И накрая, понеже (+) е непрекъснато, и композицията запазва непрекъснатостта, то също така и
$$(\texttt{+}) \circ ([\![\tau_1]\!](\overline{\varphi}) \times [\![\tau_2]\!](\overline{\varphi}))$$
  е непрекъснато изображение.

- Нека $\tau \equiv \tau_1 - \tau_2$. Ясно е, че
$$[\![\tau_1 \ \texttt{-}\ \tau_2]\!](\overline{\varphi}) = (\texttt{-}) \circ ([\![\tau_1]\!](\overline{\varphi}) \times [\![\tau_2]\!](\overline{\varphi})).$$

- Нека $\tau \equiv \tau_1 \texttt{==} \tau_2$. Ясно е, че
$$[\![\tau_1 \texttt{==} \tau_2]\!](\overline{\varphi}) = (\texttt{==}) \circ ([\![\tau_1]\!](\overline{\varphi}) \times [\![\tau_2]\!](\overline{\varphi})).$$

- Нека $\tau \equiv \texttt{if}\ \tau_0\ \texttt{then}\ \tau_1\ \texttt{else}\ \tau_2$. Ясно е, че
$$[\![\texttt{if}\ \tau_1\ \texttt{then}\ \tau_2\ \texttt{else}\ \tau_3]\!](\overline{\varphi}) = \texttt{if}_{\mathbb{N}_\perp} \circ ([\![\tau_1]\!](\overline{\varphi}) \times [\![\tau_2]\!](\overline{\varphi}) \times [\![\tau_3]\!](\overline{\varphi})).$$

- Нека $\tau \equiv \mathtt{f}_i(\tau_1,\dots,\tau_{m_i})$. Ясно е, че
$$[\![\mathtt{f}_i(\tau_1,\dots,\tau_{m_i})]\!](\overline{\varphi}) \overset{\text{деф}}{=} \varphi_i \circ ([\![\tau_1]\!](\overline{\varphi}) \times \dots \times [\![\tau_{m_i}]\!](\overline{\varphi})).$$

$\square$

Лема 3.2. Нека $\tau[\mathtt{f}_1, \dots, \mathtt{f}_k, \mathtt{x}_1, \dots, \mathtt{x}_n]$ е произволен терм. Да разгледаме произволна верига $(\overline{\varphi}_r)_{r=0}^{\infty}$ от елементи на областта на Скот $[\mathbb{N}_\perp^{m_1} \overset{\text{н}}{\to} \mathbb{N}_\perp] \times \dots \times [\mathbb{N}_\perp^{m_k} \overset{\text{н}}{\to} \mathbb{N}_\perp]$. Тогава $[\![\tau]\!]$ е непрекъснато изображение, т.е.

$$[\![\tau]\!](\bigsqcup_r \overline{\varphi}_r) = \bigsqcup_r [\![\tau]\!](\overline{\varphi}_r).$$

Да напомним, че $m_i \overset{\text{деф}}{=} \sharp\mathtt{f}_i$

Доказателство. Индукция по построението на терма τ.

- Нека $\tau \equiv \mathtt{a}$. Този случай е очевиден.
- Нека $\tau \equiv \mathtt{x}_i$.
- Нека $\tau \equiv \tau_1 + \tau_2$. Ще използваме, че $\mathtt{(+)}$ е непрекъснато изображение. Тогава:

$$\begin{aligned}
[\![\tau]\!](\bigsqcup_r \overline{\varphi}_r) &= \mathtt{(+)} \circ ([\![\tau_1]\!](\bigsqcup_r \overline{\varphi}_r) \times [\![\tau_2]\!](\bigsqcup_r \overline{\varphi}_r)) && \text{// от деф.}\\
&= \mathtt{(.)}(\mathtt{(+)}, \mathtt{cross}([\![\tau_1]\!](\bigsqcup_r \overline{\varphi}_r), [\![\tau_2]\!](\bigsqcup_r \overline{\varphi}_r)))\\
&= \mathtt{(.)}(\mathtt{(+)}, \mathtt{cross}(\bigsqcup_r [\![\tau_1]\!](\overline{\varphi}_r), \bigsqcup_r [\![\tau_2]\!](\overline{\varphi}_r))) && \text{// \textbf{(И.П.)}}\\
&= \mathtt{(.)}(\mathtt{(+)}, \bigsqcup_r \mathtt{cross}([\![\tau_1]\!](\overline{\varphi}_r), [\![\tau_2]\!](\overline{\varphi}_r))) && \text{// } \mathtt{cross} \text{ е непр.}\\
&= \bigsqcup_r \mathtt{(.)}(\mathtt{(+)}, \mathtt{cross}([\![\tau_1]\!](\overline{\varphi}_r), [\![\tau_2]\!](\overline{\varphi}_r))) && \text{// } \mathtt{(.)} \text{ е непр.}\\
&= \bigsqcup_r [\![\tau]\!](\overline{\varphi}_r). && \text{// от деф.}
\end{aligned}$$

- Нека $\tau \equiv \tau_1 \mathtt{-} \tau_2$. Използвайте, че $\mathtt{(-)}$ е непрекъснато изображение според *Задача* 1.6.
- Нека $\tau \equiv \tau_1 \mathtt{==} \tau_2$. Използвайте, че $\mathtt{(==)}$ е непрекъснато изображение според *Задача* 1.6.
- Нека $\tau \equiv \mathtt{if}\ \tau_0\ \mathtt{then}\ \tau_1\ \mathtt{else}\ \tau_2$. Използвайте, че $\mathtt{if}_{\mathbb{N}_\perp}$ е непрекъснато според *Задача* 1.9.
- Нека $\tau \equiv \mathtt{f}_i(\tau_1, \dots, \tau_{m_i})$. Тук ще използваме, че $\mathtt{(.)}$ и $\mathtt{cross}$ са непрекъснати според *Задача* 1.10 и *Задача* 1.12.

  Тогава:

$$[\![\tau]\!](\bigsqcup_n \overline{\varphi}_n) = [\![\mathtt{f}_i(\tau_1, \dots, \tau_{m_i})]\!](\bigsqcup_n \overline{\varphi}_n) \qquad \text{// } \overline{\varphi}_n = (\varphi_n^1, \dots, \varphi_n^k)$$

$$
\begin{aligned}
&= (\bigsqcup_n \varphi_n^i) \circ (\llbracket\tau_1\rrbracket(\bigsqcup_r \overline{\varphi}_r) \times \cdots \times \llbracket\tau_{m_i}\rrbracket(\bigsqcup_r \overline{\varphi}_r)) && \text{// деф. на } \llbracket\tau\rrbracket \\
&= \texttt{(.)}(\bigsqcup_n \varphi_n^i, \texttt{cross}(\llbracket\tau_1\rrbracket(\bigsqcup_r \overline{\varphi}_r), \dots, \llbracket\tau_{m_i}\rrbracket(\bigsqcup_r \overline{\varphi}_r)) \\
&= \texttt{(.)}(\bigsqcup_n \varphi_n^i, \texttt{cross}(\bigsqcup_r \llbracket\tau_1\rrbracket(\overline{\varphi}_r), \dots, \bigsqcup_r \llbracket\tau_{m_i}\rrbracket(\overline{\varphi}_r)) && \text{// от } \textbf{(И.П.)} \\
&= \texttt{(.)}(\bigsqcup_n \varphi_n^i, \bigsqcup_r \texttt{cross}(\llbracket\tau_1\rrbracket(\overline{\varphi}_r), \dots, \llbracket\tau_{m_i}\rrbracket(\overline{\varphi}_r)) && \text{// } \texttt{cross} \text{ е непр.} \\
&= \bigsqcup_r \texttt{(.)}(\varphi_r^i, \texttt{cross}(\llbracket\tau_1\rrbracket(\overline{\varphi}_r), \dots, \llbracket\tau_{m_i}\rrbracket(\overline{\varphi}_r))) && \text{// } \texttt{(.)} \text{ е непр.} \\
&= \bigsqcup_r \varphi_r^i(\llbracket\tau_1\rrbracket(\overline{\varphi}_r), \dots, \llbracket\tau_{m_i}\rrbracket(\overline{\varphi}_r)) \\
&= \bigsqcup_r \llbracket \texttt{f}_i(\tau_1, \dots, \tau_{m_i})\rrbracket(\overline{\varphi}_r) \\
&= \bigsqcup_r \llbracket\tau\rrbracket(\overline{\varphi}_r).
\end{aligned}
$$

□

Оттук нататък, можем да сме сигурни, че ако едно изображение може да се дефинира като терм на езика `FUN` , то със сигурност това изображение е непрекъснато.

Теорема 3.1. За всеки терм $\tau[\texttt{f}_1, \dots, \texttt{f}_k, \texttt{x}_1, \dots, \texttt{x}_n]$, имаме, че

$$\llbracket\tau\rrbracket \in [[\mathbb{N}_\perp^{m_1} \xrightarrow{\text{H}} \mathbb{N}_\perp] \times \cdots \times [\mathbb{N}_\perp^{m_k} \xrightarrow{\text{H}} \mathbb{N}_\perp] \xrightarrow{\text{H}} [\mathbb{N}_\perp^n \xrightarrow{\text{H}} \mathbb{N}_\perp]].$$

Нека е дадена една рекурсивна програма $\texttt{P}[\bar{\texttt{f}}, \bar{\texttt{x}}]$, където:

Можем да си мислим, че $\texttt{f}_1$ е нещо като `main` функция за програмата `P`.

$$
\texttt{P} = \begin{cases}
\texttt{f}_1(\texttt{x}_1, \dots, \texttt{x}_{m_1}) = \tau_1[\texttt{f}_1, \dots, \texttt{f}_k, \texttt{x}_1, \dots, \texttt{x}_{m_1}] \\
\texttt{f}_2(\texttt{x}_1, \dots, \texttt{x}_{m_2}) = \tau_2[\texttt{f}_1, \dots, \texttt{f}_k, \texttt{x}_1, \dots, \texttt{x}_{m_2}] \\
\vdots \\
\texttt{f}_k(\texttt{x}_1, \dots, \texttt{x}_{m_k}) = \tau_k[\texttt{f}_1, \dots, \texttt{f}_k, \texttt{x}_1, \dots, \texttt{x}_{m_k}]
\end{cases}
$$

Нека $\overline{\gamma} \in [\mathbb{N}_\perp^{m_1} \xrightarrow{\text{H}} \mathbb{N}_\perp] \times \cdots \times [\mathbb{N}_\perp^{m_k} \xrightarrow{\text{H}} \mathbb{N}_\perp]$ е *най-малкото решение* на системата

В тази система неизвестните са $\varphi_1, \dots, \varphi_k$.

$$
\left|
\begin{array}{rcl}
\varphi_1 & = & \llbracket\tau_1\rrbracket(\varphi_1, \dots, \varphi_k) \\
& \vdots & \\
\varphi_k & = & \llbracket\tau_k\rrbracket(\varphi_1, \dots, \varphi_k).
\end{array}
\right.
$$

От Теоремата на Клини знаем, че такова най-малко решение съществува.

За дадената рекурсивна програма $\mathtt{P}[\bar{\mathtt{f}}, \bar{\mathtt{x}}]$, определяме **денотационната семантика с предаване на параметрите по име** като изображението $\mathcal{D}[\![\mathtt{P}]\!] \in [\mathbb{N}_{\perp}^{m_1} \xrightarrow{\text{н}} \mathbb{N}_{\perp}]$, където:

$$\mathcal{D}[\![\mathtt{P}]\!](a_1, \ldots, a_{m_1}) \stackrel{\text{деф}}{=} \begin{cases} \gamma_1(a_1, \ldots, a_{m_1}), & \text{ако } \perp \notin \{a_1, \ldots, a_{m_1}\} \\ \perp, & \text{ако } \perp \in \{a_1, \ldots, a_{m_1}\}. \end{cases}$$

Пример 3.4. Да разгледаме следната проста програма:

```
f(x) = if x == 0 then 1 else f(x+1)
```

Тази програма е дори валидна програма на хаскел.

На тази програма съответства системата от едно уравнение с една функционална променлива

$$f = \Gamma(f), \tag{3.2}$$

където за изображението Γ имаме, че:

$$\Gamma \in [[\mathbb{N}_{\perp} \xrightarrow{\text{н}} \mathbb{N}_{\perp}] \xrightarrow{\text{н}} [\mathbb{N}_{\perp} \xrightarrow{\text{н}} \mathbb{N}_{\perp}]]$$
$$\Gamma(f)(x) \stackrel{\text{деф}}{=} \begin{cases} 1, & \text{ако } x = 0 \\ f(x+1), & \text{иначе}. \end{cases}$$

Най-малкото решение на системата (3.2) е функцията $\varphi : \mathbb{N}_{\perp} \to \mathbb{N}_{\perp}$, за която

$$\varphi(x) = \begin{cases} 1, & \text{ако } x = 0 \\ \perp, & \text{иначе}. \end{cases}$$

Обаче системата (3.2) има и други решения. Например, функцията $\psi(x) = 1$ за всяко $x \in \mathbb{N}_{\perp}$. Този пример ни показва, че най-малкото решение кореспондира точно с нашата интуиция как „работи” тази програма.

✍ Съобразете, че системата (3.2) има дори безкрайно много решения!

3.3 Операционна семантика

Това се нарича cost dynamics в [8, стр. 58].

Дефинираме релация $\mu \Downarrow^{\ell}_{\mathtt{P}} \mathtt{a}$, която ни казва как един функционален терм μ се свежда до константата $\mathtt{a}$ след ℓ на брой стъпки следвайки следните правила:

$$\frac{}{\mathtt{a} \Downarrow^{0}_{\mathtt{P}} \mathtt{a}}\ (\text{const})$$

$$\frac{\mu_1 \Downarrow^{\ell_1}_{\mathtt{P}} \mathtt{a}_1 \quad \mu_2 \Downarrow^{\ell_2}_{\mathtt{P}} \mathtt{a}_2 \quad a = \mathtt{(+)}(a_1, a_2)}{\mu_1 + \mu_2 \Downarrow^{\ell_1+\ell_2+1}_{\mathtt{P}} \mathtt{a}}\ (\text{plus})$$

$$\frac{\mu_1 \Downarrow^{\ell_1}_{\mathtt{P}} \mathtt{a}_1 \quad \mu_2 \Downarrow^{\ell_2}_{\mathtt{P}} \mathtt{a}_2 \quad a = \mathtt{(-)}(a_1, a_2)}{\mu_1 - \mu_2 \Downarrow^{\ell_1+\ell_2+1}_{\mathtt{P}} \mathtt{a}}\ (\text{minus})$$

$$\frac{\mu_1 \Downarrow^{\ell_1}_{\mathtt{P}} \mathtt{a}_1 \quad \mu_2 \Downarrow^{\ell_2}_{\mathtt{P}} \mathtt{a}_2 \quad a = \mathtt{(==)}(a_1, a_2)}{\mu_1 \mathtt{==} \mu_2 \Downarrow^{\ell_1+\ell_2+1}_{\mathtt{P}} \mathtt{a}}\ (\text{eq})$$

$$\frac{\mu_0 \Downarrow^{\ell_0}_{\mathtt{P}} \mathtt{a}_0 \quad \mu_1 \Downarrow^{\ell_1}_{\mathtt{P}} \mathtt{a}_1 \quad \mathtt{a}_0 \not\equiv \mathtt{0}}{\mathtt{if}\ \mu_0\ \mathtt{then}\ \mu_1\ \mathtt{else}\ \mu_2 \Downarrow^{\ell_0+\ell_1+1}_{\mathtt{P}} \mathtt{a}_1}\ (\text{if}^{+})$$

$$\frac{\mu_0 \Downarrow^{\ell_0}_{\mathtt{P}} \mathtt{0} \quad \mu_2 \Downarrow^{\ell_2}_{\mathtt{P}} \mathtt{a}_2}{\mathtt{if}\ \mu_0\ \mathtt{then}\ \mu_1\ \mathtt{else}\ \mu_2 \Downarrow^{\ell_0+\ell_2+1}_{\mathtt{P}} \mathtt{a}_2}\ (\text{if}_0)$$

$$\frac{\tau_i[\mathtt{x}_1/\mu_1, \ldots, \mathtt{x}_{m_i}/\mu_{m_i}] \Downarrow^{\ell}_{\mathtt{P}} \mathtt{a}}{\mathtt{f}_i(\mu_1, \ldots, \mu_{m_i}) \Downarrow^{\ell+1}_{\mathtt{P}} \mathtt{a}}\ (\text{cbn})$$

Фигура 3.1: Правила на операционната семантика на `FUN`

Забележка 3.1. Защо нямаме следното правило?

$$\frac{\mu_0 \Downarrow^{\ell_0}_{\mathtt{P}} \mathtt{a}_0 \quad \mu_1 \Downarrow^{\ell_1}_{\mathtt{P}} \mathtt{a}_1 \quad \mu_2 \Downarrow^{\ell_2}_{\mathtt{P}} \mathtt{a}_2 \quad a = \mathtt{if}(a_0, a_1, a_2)}{\mathtt{if}\ \mu_0\ \mathtt{then}\ \mu_1\ \mathtt{else}\ \mu_2 \Downarrow^{\ell_0+\ell_1+\ell_2+1}_{\mathtt{P}} \mathtt{a}}$$

Забележка 3.2. Ако искахме да дефинираме операционна семантика с предаване на параметрите по стойност, то щяхме да заменим правилото (cbn) със следното правило:

$$(\ell = \textstyle\sum_{k\leq m_i} \ell_k)\ \dfrac{\tau_i[\mathtt{x}_1/\mathtt{b}_1,\ldots,\mathtt{x}_{m_i}/\mathtt{b}_{m_i}] \Downarrow_{\mathtt{P}}^{\ell_0} \mathtt{a} \quad \mu_1 \Downarrow_{\mathtt{P}}^{\ell_1} \mathtt{b}_1 \quad \cdots \quad \mu_{m_i} \Downarrow_{\mathtt{P}}^{\ell_{m_i}} \mathtt{b}_{m_i}}{\mathtt{f}_i(\mu_1,\ldots,\mu_{m_i}) \Downarrow_{\mathtt{P}}^{\ell+1} \mathtt{a}}\ \text{(cbv)}$$

Ще пишем $\mu \Downarrow_{\mathtt{P}} \mathtt{a}$, ако съществува ℓ, за което $\mu \Downarrow_{\mathtt{P}}^{\ell} \mathtt{a}$. Също така, понякога ще пишем $\mu \Downarrow_{\mathtt{P}}^{<\ell} \mathtt{a}$, когато искаме да кажем, че μ се свежда до $\mathtt{a}$ след прилагане на по-малко от ℓ на брой правила от операционната семантика.

Лема 3.3. Докажете, че за всеки затворен терм μ, ако $\mu \Downarrow_{\mathtt{P}} \mathtt{a}$ и $\mu \Downarrow_{\mathtt{P}} \mathtt{a}'$, то $\mathtt{a} \equiv \mathtt{a}'$.

Защо не можем да направим индукция по построението на термовете?

Упътване. Индукция по дължината на изчислението. □

За фиксирана програма $\mathtt{P}$, нека за всеки *функционален* терм μ да дефинираме

$$\mathrm{eval}_{\mathtt{P}}[\![\mu]\!] \stackrel{\text{деф}}{=} \begin{cases} b, & \text{ако } \mu \Downarrow_{\mathtt{P}} \mathtt{b} \\ \bot, & \text{ако } \mu \text{ няма извод до константа.} \end{cases}$$

Операционната семантика по име на рекурсивната програма $\mathtt{P}[\bar{\mathtt{x}}, \bar{\mathtt{f}}]$ представлява изображението

$$\mathcal{O}[\![\mathrm{P}]\!] \in [\mathbb{N}_\bot^{m_1} \xrightarrow{\text{н}} \mathbb{N}_\bot],$$

където

$$\mathcal{O}[\![\mathrm{P}]\!](a_1,\ldots,a_{m_1}) \stackrel{\text{деф}}{=} \begin{cases} \mathrm{eval}_{\mathtt{P}}[\![\mathtt{f}_1(\mathtt{a}_1,\ldots,\mathtt{a}_{m_1})]\!], & \text{ако } \bot \notin \{a_1,\ldots,a_{m_1}\} \\ \bot, & \text{ако } \bot \in \{a_1,\ldots,a_{m_1}\} \end{cases}$$

за произволни $a_1,\ldots,a_{m_1} \in \mathbb{N}_\bot$.

Забележка 3.3. Всъщност ние все още няма как да знаем, че за всяка програма $\mathtt{P}$, $\mathcal{O}[\![\mathrm{P}]\!]$ е непрекъснато изображение. Този факт може да се докаже директно, но вместо това, ние ще видим, че $\mathcal{O}[\![\mathrm{P}]\!] = \mathcal{D}[\![\mathrm{P}]\!]$ и оттам ще получим непрекъснатостта на $\mathcal{O}[\![\mathrm{P}]\!]$, защото от дефиницията на $\mathcal{D}[\![\mathrm{P}]\!]$ е ясно, че то е непрекъснато изображение.

Пример 3.5. Нека за програмата $\mathtt{P}$:

```
f(x,y) = if x == y then 0 else 1 + f(x,y+1)
```

да разгледаме няколко извода с правилата на операционната семантика по име.

$$
\dfrac{
\dfrac{
\dfrac{\dfrac{}{\mathtt{3} \Downarrow^{0}_{\mathtt{P}} \mathtt{3}}\,(\text{const}) \quad \dfrac{}{\mathtt{2} \Downarrow^{0}_{\mathtt{P}} \mathtt{2}}\,(\text{const})}{\mathtt{3 == 2} \Downarrow^{1}_{\mathtt{P}} \mathtt{0}}\,(\text{eq})
\quad
\dfrac{\dfrac{}{\mathtt{1} \Downarrow^{0}_{\mathtt{P}} \mathtt{1}}\,(\text{const}) \quad
\dfrac{\dfrac{\dfrac{\dfrac{}{\mathtt{3} \Downarrow^{0}_{\mathtt{P}} \mathtt{3}}\,(\text{const}) \quad \dfrac{\dfrac{}{\mathtt{1} \Downarrow^{0}_{\mathtt{P}} \mathtt{1}}\,(\text{const}) \quad \dfrac{}{\mathtt{2} \Downarrow^{0}_{\mathtt{P}} \mathtt{2}}\,(\text{const})}{\mathtt{2+1} \Downarrow^{1}_{\mathtt{P}} \mathtt{3}}\,(\text{plus})}{\mathtt{3 == 2 + 1} \Downarrow^{2}_{\mathtt{P}} \mathtt{1}}\,(\text{eq}) \quad \dfrac{}{\mathtt{0} \Downarrow^{0}_{\mathtt{P}} \mathtt{0}}\,(\text{const})}{\mathtt{if\ 3\ ==\ 2+1\ then\ 0\ else\ 1+f(3,2+1+1)} \Downarrow^{3}_{\mathtt{P}} \mathtt{0}}\,(\text{if}^{+})}{\mathtt{f(3,2+1)} \Downarrow^{4}_{\mathtt{P}} \mathtt{0}}\,(\text{cbn})
}{\mathtt{1 + f(3,2+1)} \Downarrow^{5}_{\mathtt{P}} \mathtt{1}}\,(\text{plus})
}{\mathtt{if\ 3\ ==\ 2\ then\ 0\ else\ 1\ +\ f(3,2+1)} \Downarrow^{7}_{\mathtt{P}} \mathtt{1}}\,(\text{if}_0)
}{\mathtt{f(3,2)} \Downarrow^{8}_{\mathtt{P}} \mathtt{1}}\,(\text{cbn})
$$

(а) Крайно дърво на извод започващо от функционалния терм `f(3,2)`

$$
\dfrac{
\dfrac{
\dfrac{\dfrac{}{\mathtt{2} \Downarrow^{0}_{\mathtt{P}} \mathtt{2}}\,(\text{const}) \quad \dfrac{}{\mathtt{3} \Downarrow^{0}_{\mathtt{P}} \mathtt{3}}\,(\text{const})}{\mathtt{2 == 3} \Downarrow^{1}_{\mathtt{P}} \mathtt{0}}\,(\text{eq})
\quad
\dfrac{\dfrac{}{\mathtt{1} \Downarrow^{0}_{\mathtt{P}} \mathtt{1}}\,(\text{const}) \quad
\dfrac{\dfrac{\dfrac{\dfrac{}{\mathtt{2} \Downarrow^{0}_{\mathtt{P}} \mathtt{2}}\,(\text{const}) \quad \dfrac{\dfrac{}{\mathtt{3} \Downarrow^{0}_{\mathtt{P}} \mathtt{3}}\,(\text{const}) \quad \dfrac{}{\mathtt{1} \Downarrow^{0}_{\mathtt{P}} \mathtt{1}}\,(\text{const})}{\mathtt{3+1} \Downarrow^{1}_{\mathtt{P}} \mathtt{4}}\,(\text{plus})}{\mathtt{2 == 3+1} \Downarrow^{2}_{\mathtt{P}} \mathtt{0}}\,(\text{eq}) \quad \dfrac{\vdots}{\mathtt{1 + f(2,3+1+1)} \Downarrow^{\ell}_{\mathtt{P}} \square}}{\mathtt{if\ 2\ ==\ 3+1\ then\ 0\ else\ 1\ +\ f(2,3+1+1)} \Downarrow^{\ell+3}_{\mathtt{P}} \square}\,(\text{if}_0)}{\mathtt{f(2,3+1)} \Downarrow^{\ell+4}_{\mathtt{P}} \square}\,(\text{cbn})
}{\mathtt{1+f(2,3+1)} \Downarrow^{\ell+5}_{\mathtt{P}} \square}\,(\text{plus})
}{\mathtt{if\ 2\ ==\ 3\ then\ 0\ else\ 1\ +\ f(2,3+1)} \Downarrow^{\ell+6}_{\mathtt{P}} \square}\,(\text{if}_0)
}{\mathtt{f(2,3)} \Downarrow^{\ell+7}_{\mathtt{P}} \square}\,(\text{cbn})
$$

(б) Част от безкрайното дърво на извод започващо от функционалния терм `f(2,3)`

3.4 Теорема за еквивалентност

Винаги с μ ще означаваме функционални термове, а с τ произволни термове.

> **Твърдение 3.2.** Да разгледаме една програма $\mathtt{P}$. Нека $\mu[\mathtt{f}_1, \dots, \mathtt{f}_n]$ е *функционален* терм. Тогава
>
> $$\text{eval}_{\mathtt{P}}[\![\mu]\!] \sqsubseteq [\![\mu]\!](\bar{\gamma}),$$
>
> където $\bar{\gamma} = (\gamma_1, \dots, \gamma_n)$ е някоя неподвижна точка на непрекъснатия оператор
>
> $$\Gamma \stackrel{\text{деф}}{=} [\![\tau_1]\!] \times \cdots \times [\![\tau_n]\!],$$
>
> който съответства на програмата $\mathtt{P}$.

Доказателство. Ако от терма μ *няма извод* до константа, то по дефиниция $\text{eval}_{\mathtt{P}}[\![\mu]\!] = \bot$ и в този случай е очевидно, че

$$\text{eval}_{\mathtt{P}}[\![\mu]\!] \sqsubseteq [\![\mu]\!](\bar{\gamma}).$$

В [14, стр. 192], [12, стр. 157] доказателството е друго. Тук използваме наготово *Задача* 1.11. За това твърдение не е необходимо $\bar{\gamma}$ да бъде най-малката неподвижна точка на Γ, а просто решение. Само за другата посока ще е важно да е най-малкото решение.

Интересният случай е когато от терма μ *има извод* до константа. Тогава ще докажем, че за произволен функционален терм μ и константа $\mathtt{a}$,

$$\mu \Downarrow_{\mathtt{P}} \mathtt{a} \implies [\![\mu]\!](\bar{\gamma}) = a. \tag{3.3}$$

Доказателството на (3.3) ще проведем с *пълна* индукция по дължината ℓ на извода $\mu \Downarrow_{\mathtt{P}}^{\ell} \mathtt{a}$.

Константата $\mathtt{a}$ има стойност числото a.

Нека изводът има дължина 0. Понеже μ е функционален терм, според правилата на операционната семантика, единствената възможност е $\mu \equiv \mathtt{a}$. Тогава е очевидно, че $[\![\mu]\!](\bar{\gamma}) = a$. Ясно е, че в този случай,

$$\mathtt{a} \Downarrow_{\mathtt{P}}^{0} \mathtt{a} \implies [\![\mathtt{a}]\!](\bar{\gamma}) = a.$$

Нака имаме следното индукционно предположение:

$$\mu \Downarrow_{\mathtt{P}}^{<\ell} \mathtt{a} \implies [\![\mu]\!](\bar{\gamma}) = a. \tag{3.4}$$

Ще докажем, че

$$\mu \Downarrow_{\mathtt{P}}^{\ell} \mathtt{a} \implies [\![\mu]\!](\bar{\gamma}) = a.$$

Ще разгледаме всички случаи в зависимост от вида на функционалния терм μ.

- Да разгледаме случая, когато $\mu \equiv \mu_1 + \mu_2$. Нека $\mu_1 + \mu_2 \Downarrow_{\mathtt{P}}^{\ell} \mathtt{a}$. Според правилата за извод в операционната семантика по име, имаме следната ситуация:

$$\dfrac{\dfrac{\vdots}{\mu_1 \Downarrow_{\mathtt{P}}^{\ell_1} \mathtt{a}_1} \qquad \dfrac{\vdots}{\mu_2 \Downarrow_{\mathtt{P}}^{\ell_2} \mathtt{a}_2}}{\mu_1 + \mu_2 \Downarrow_{\mathtt{P}}^{\ell_1+\ell_2+1} \mathtt{a}} \text{ правило (plus)}$$

където $a = \texttt{(+)}(a_1, a_2)$. Ясно е, че изводите на $\mu_1 \Downarrow_{\mathtt{P}}^{<\ell} \mathtt{a}_1$ и $\mu_2 \Downarrow_{\mathtt{P}}^{<\ell} \mathtt{a}_2$. Следователно можем да приложим **(И.П.)** за μ_1 и μ_2, откъдето получаваме, че

$$\begin{aligned} \mu_1 \Downarrow_{\mathtt{P}}^{<\ell} \mathtt{a}_1 &\implies [\![\mu_1]\!](\overline{\gamma}) = \mathtt{a}_1 \\ \mu_2 \Downarrow_{\mathtt{P}}^{<\ell} \mathtt{a}_2 &\implies [\![\mu_2]\!](\overline{\gamma}) = \mathtt{a}_2. \end{aligned}$$

Тогава получаваме, че ако $\mu_1 + \mu_2 \Downarrow_{\mathtt{P}}^{\ell} \mathtt{a}$, то

$$\begin{aligned} [\![\mu_1 + \mu_2]\!](\overline{\gamma}) &\stackrel{\text{деф}}{=} \texttt{(+)}([\![\mu_1]\!](\overline{\gamma}), [\![\mu_2]\!](\overline{\gamma})) \\ &= \texttt{(+)}(a_1, a_2) \\ &= a. \end{aligned}$$

- Случаят, когато $\mu \equiv \mu_1$ - μ_2 е лесен.
- Случаят, когато $\mu \equiv \mu_1$ == μ_2 е лесен.
- Случаят, когато $\mu \equiv \texttt{if}\ \mu_0\ \texttt{then}\ \mu_1\ \texttt{else}\ \mu_2$, е лесен.
- Нека имаме функционалния терм $\mu \equiv \mathtt{f}_i(\mu_1, \ldots, \mu_{m_i})$ и $\mu \Downarrow_{\mathtt{P}}^{\ell} \mathtt{a}$. Според правилата за извод в операционната семантика по име, имаме следната ситуация:

$$\dfrac{\dfrac{\vdots}{\tau_i[\mathtt{x}_1/\mu_1, \ldots, \mathtt{x}_{m_i}/\mu_{m_i}] \Downarrow_{\mathtt{P}}^{\ell-1} \mathtt{a}}}{\mathtt{f}_i(\mu_1, \ldots, \mu_{m_i}) \Downarrow_{\mathtt{P}}^{\ell} \mathtt{a}} \text{ правило (cbn)}$$

От **(И.П.)** имаме, че

$$\tau_i[\overline{\mathtt{x}}/\overline{\mu}] \Downarrow_{\mathtt{P}}^{<\ell} \mathtt{a} \implies [\![\tau_i[\overline{\mathtt{x}}/\overline{\mu}]]\!](\overline{\gamma}) = a.$$

Понеже знаем, че $\tau_i[\overline{\mathtt{x}}/\overline{\mu}] \Downarrow_{\mathtt{P}}^{<\ell} \mathtt{a}$, то $[\![\tau_i[\overline{\mathtt{x}}/\overline{\mu}]]\!](\overline{\gamma}) = a$.

Лесно се съобразява, че:

$$\begin{aligned} [\![\tau_i[\overline{\mathtt{x}}/\overline{\mu}]]\!](\overline{\gamma}) &= [\![\tau_i]\!](\overline{\gamma})([\![\mu_1]\!](\overline{\gamma}), \ldots, [\![\mu_{m_i}]\!](\overline{\gamma})) && \text{// Лема за замяната} \\ &= \gamma_i([\![\mu_1]\!](\overline{\gamma}), \ldots, [\![\mu_{m_i}]\!](\overline{\gamma})) && \text{// } \gamma_i = [\![\tau_i]\!](\overline{\gamma}) \\ &= [\![\mathtt{f}_i(\mu_1, \ldots, \mu_{m_i})]\!](\overline{\gamma}). && \text{// деф. на стойност на терм} \end{aligned}$$

Обединявайки всичко, което знаем, получаваме:

$$\begin{aligned} \mathtt{f}_i(\mu_1, \ldots, \mu_{m_i}) \Downarrow_{\mathtt{P}}^{\ell} \mathtt{a} &\implies \tau_i[\overline{\mathtt{x}}/\overline{\mu}] \Downarrow_{\mathtt{P}}^{<\ell} \mathtt{a} && \text{// правило (cbn)} \\ &\implies [\![\tau_i[\overline{\mathtt{x}}/\overline{\mu}]]\!](\overline{\gamma}) = a && \text{// \textbf{(И.П.)}} \\ &\implies [\![\mathtt{f}_i(\mu_1, \ldots, \mu_{m_i})]\!](\overline{\gamma}) = a. \end{aligned}$$

Индукционното предположение е цялата импликация. Според случая, който разглеждаме и според правилата на операционната семантика, лявата страна на импликацията е изпълнена. Следователно, и дясната страна на импликацията е изпълнена.
Озн. $\overline{\gamma} = (\gamma_1, \ldots, \gamma_n)$. Единствено в този случай се използва, че $\overline{\gamma}$ е решение на системата от оператори, като не е задължително да е най-малкото решение.

С това доказателството на (3.3) е завършено. □

Следствие 3.1 (Теорема за коректност)**.** За всяка рекурсивна програма P на езика FUN имаме, че

$$\mathcal{O}[\![\mathtt{P}]\!] \sqsubseteq \mathcal{D}[\![\mathtt{P}]\!].$$

Доказателство. Нека $\overline{\gamma}$ е най-малкото решение на системата от непрекъснати оператори, която съответства на програмата P. Тогава за произволни $a_1, \ldots, a_{m_1} \in \mathbb{N}$,

$$\begin{aligned}
\mathcal{O}[\![\mathtt{P}]\!](a_1, \ldots, a_{m_1}) &= \mathrm{eval}_{\mathtt{P}}[\![\mathtt{f}_1(\mathtt{a}_1, \ldots, \mathtt{a}_{m_1})]\!] && \text{// от деф.}\\
&\sqsubseteq [\![\mathtt{f}_1(\mathtt{a}_1, \ldots, \mathtt{a}_{m_1})]\!](\overline{\gamma}) && \text{// \textit{Твърдение} 3.2}\\
&= \gamma_1([\![\mathtt{a}_1]\!](\overline{\gamma}), \ldots, [\![\mathtt{a}_{m_1}]\!](\overline{\gamma})) && \text{// стойност на терм}\\
&= \gamma_1(a_1, \ldots, a_{m_1}) && \text{// } [\![\mathtt{a}_i]\!](\overline{\gamma}) = a_i\\
&= \mathcal{D}[\![\mathtt{P}]\!](a_1, \ldots, a_{m_1}). && \text{// от деф.}
\end{aligned}$$

□

Така получихме едната посока на теоремата за еквивалентност. Сега преминаваме към доказателството на другата посока.

Нека сега $\overline{\gamma} = (\gamma_1, \ldots, \gamma_k)$ е най-малкото решение на системата от оператори, която съответства на програма P за денотационната семантика по име. Да напомним, че това означава, че $\overline{\gamma} = \bigsqcup_r \overline{\gamma}_r$, където $\overline{\gamma}_0 = (\bot^{(m_1)}, \ldots, \bot^{(m_k)})$ и за всяко r,

$$\begin{aligned}
\gamma^i_{r+1} &= [\![\tau_i]\!](\overline{\gamma}_r)\\
\overline{\gamma}_r &= (\gamma^1_r, \ldots, \gamma^k_r).
\end{aligned}$$

Твърдение 3.3. Да разгледаме една програма P в езика FUN , като $\overline{\gamma} = \bigsqcup_r \overline{\gamma}_r$ е най-малкото решение на системата от оператори за P. Тогава за произволен терм $\tau[\mathtt{f}_1, \ldots, \mathtt{f}_k, \mathtt{x}_1, \ldots, \mathtt{x}_n]$ и произволни функционални термове

$$\mu_1[\mathtt{f}_1, \ldots, \mathtt{f}_k], \ldots, \mu_n[\mathtt{f}_1, \ldots, \mathtt{f}_k],$$

и всяко r, е изпълнено, че:

$$[\![\tau]\!](\overline{\gamma}_r)(\mathrm{eval}_{\mathtt{P}}[\![\mu_1]\!], \ldots, \mathrm{eval}_{\mathtt{P}}[\![\mu_n]\!]) \sqsubseteq \mathrm{eval}_{\mathtt{P}}[\![\tau[\overline{\mathtt{x}}/\overline{\mu}]]\!].$$

Доказателство. За произволно естествено число r, нека твърдението $\mathtt{Include}(r)$ да гласи следното: „за произволен терм $\tau[\mathtt{f}_1, \ldots, \mathtt{f}_k, \mathtt{x}_1, \ldots, \mathtt{x}_n]$ и произволни

✍ На пръв поглед не е ясно защо трябва да докажем толкова сложно формулирано твърдение.Съобразете защо не можем да докажем по-простото твърдение, че за всеки функционален терм μ и всяко r, $[\![\mu]\!](\overline{\gamma}_r) \sqsubseteq \mathrm{eval}_{\mathtt{P}}[\![\mu]\!]$.

функционални термове $\mu_1[\mathtt{f}_1, \ldots, \mathtt{f}_k], \ldots, \mu_n[\mathtt{f}_1, \ldots, \mathtt{f}_k]$, е изпълнено, че:

$$[\![\tau]\!](\overline{\gamma}_r)(\mathrm{eval}_{\mathtt{P}}[\![\mu_1]\!], \ldots, \mathrm{eval}_{\mathtt{P}}[\![\mu_n]\!]) \sqsubseteq \mathrm{eval}_{\mathtt{P}}[\![\tau[\bar{\mathtt{x}}/\overline{\mu}]]\!].\text{”}$$

Ще докажем с индукция по r, че $\mathtt{Include}(r)$ е изпълнено за всяко r.

- Първо ще докажем $\mathtt{Include}(0)$. Това ще направим с индукция по построението на терма τ. Да разгледаме произволни функционални термове μ_i и за улеснение да положим $a_i \stackrel{\text{деф}}{=} \mathrm{eval}_{\mathtt{P}}[\![\mu_i]\!]$.
  - Нека $\tau \equiv \mathtt{c}$. Тогава:

$$\begin{aligned} [\![\mathtt{c}]\!](\overline{\gamma}_0)(\overline{a}) &= c && \text{// стойност на терма}\\ &= \mathrm{eval}_{\mathtt{P}}[\![\mathtt{c}]\!] && \text{// правило (1)}\\ &= \mathrm{eval}_{\mathtt{P}}[\![\mathtt{c}[\bar{\mathtt{x}}/\overline{\mu}]]\!]. \end{aligned}$$

  - Нека $\tau \equiv \mathtt{x}_i$. Тогава:

$$\begin{aligned} [\![\mathtt{x}_i]\!](\overline{\gamma}_0)(\overline{a}) &= a_i && \text{// стойност на терма}\\ &= \mathrm{eval}_{\mathtt{P}}[\![\mu_i]\!] && \text{// } a_i \stackrel{\text{деф}}{=} \mathrm{eval}_{\mathtt{P}}[\![\mu_i]\!]\\ &= \mathrm{eval}_{\mathtt{P}}[\![\mathtt{x}_i[\bar{\mathtt{x}}/\overline{\mu}]]\!]. \end{aligned}$$

  - Нека $\tau \equiv \tau_1 + \tau_2$. В този случай, τ е съставен от по-простите термове τ_1 и τ_2. От **(И.П.)** за τ_1 и τ_2 следва, че за $i = 1, 2$ е изпълнено следното:

$$[\![\tau_i]\!](\overline{\gamma}_0)(\overline{a}) \sqsubseteq \mathrm{eval}_{\mathtt{P}}[\![\tau_i[\bar{\mathtt{x}}/\overline{\mu}]]\!]. \tag{3.5}$$

    Да напомним, че изображението `(+)` е непрекъснато, откъдето следва, че също така е монотонно. Тогава:

$$\begin{aligned} [\![\tau_1 + \tau_2]\!](\overline{\gamma}_0)(\overline{a}) &= \mathtt{(+)}([\![\tau_1]\!](\overline{\gamma}_0)(\overline{a}), [\![\tau_2]\!](\overline{\gamma}_0)(\overline{a})) && \text{// стойност на терма}\\ &\sqsubseteq \mathtt{(+)}(\mathrm{eval}_{\mathtt{P}}[\![\tau_1[\bar{\mathtt{x}}/\overline{\mu}]]\!], \mathrm{eval}_{\mathtt{P}}[\![\tau_2[\bar{\mathtt{x}}/\overline{\mu}]]\!]) && \text{// от (3.5)}\\ &= \mathrm{eval}_{\mathtt{P}}[\![\tau[\bar{\mathtt{x}}/\overline{\mu}]]\!] && \text{// правило } (\mathtt{plus}) \end{aligned}$$

  - Нека $\tau \equiv \tau_1$ `-` τ_2.
  - Нека $\tau \equiv \tau_1$ `==` τ_2.
  - Нека $\tau \equiv$ `if` τ_0 `then` τ_1 `else` τ_2.
  - Нека $\tau \equiv \mathtt{f}_i(\rho_1, \ldots, \rho_{m_i})$. Тогава

✍ Тези три случая са за домашно!

$$\begin{aligned} [\![\tau]\!](\overline{\gamma}_0)(\overline{a}) &= \gamma_0^i([\![\rho_1]\!](\overline{\gamma}_0)(\overline{a}), \ldots, [\![\rho_{m_i}]\!](\overline{\gamma}_0)(\overline{a})) && \text{// стойност на терма}\\ &= \bot && \text{// } \gamma_0^i(\overline{x}) \stackrel{\text{деф}}{=} \bot\\ &\sqsubseteq \mathrm{eval}_{\mathtt{P}}[\![\tau[\bar{\mathtt{x}}/\overline{\mu}]]\!]. \end{aligned}$$

  Така доказахме, че $\mathtt{Include}(0)$ е изпълнено.

- Нека $r > 0$. Да приемем, че $\texttt{Include}(r-1)$ е изпълнено. Ще докажем $\texttt{Include}(r)$ отново с индукция по построението на термовете. Единственият случай, който заслужава внимание е

  $$\tau \equiv \texttt{f}_i(\rho_1, \ldots, \rho_{m_i}).$$

  Доказателствата на всички останали случаи за τ протичат по абсолютно същия начин както при $r = 0$.

  ✍ Разгледайте сами останалите случаи за τ и се убедете, че те наистина се доказват по същия начин както при $r = 0$.

  Понеже термът τ е построен с помощта на термовете ρ_j, за $j = 1, \ldots, m_i$, можем да приложим **(И.П.)** за тях и да получим, че

  $$\begin{aligned} [\![\rho_j]\!](\overline{\gamma}_r)(\underbrace{\text{eval}_{\text{P}}[\![\mu_1]\!]}_{a_1}, \ldots, \underbrace{\text{eval}_{\text{P}}[\![\mu_n]\!]}_{a_n}) &= \underbrace{[\![\rho_j]\!](\overline{\gamma}_r)(a_1, \ldots, a_n)}_{b_j} \\ &\sqsubseteq \underbrace{\text{eval}_{\text{P}}[\![\rho_j[\bar{\texttt{x}}/\overline{\mu}]]\!]}_{c_j}. && \text{// от } \textbf{(И.П.)} \end{aligned}$$

  Обърнете внимание, че ρ'_j са функционални термове

  Нека за наше улеснение да положим $\rho'_j \stackrel{\text{деф}}{=} \rho_j[\bar{\texttt{x}}/\overline{\mu}]$. Това означава, че до момента имаме следното:

  $$[\![\tau_i]\!](\overline{\varphi})(\underbrace{[\![\rho_1]\!](\overline{\gamma}_r)(\overline{a})}_{b_1}, \ldots, \underbrace{[\![\rho_{m_i}]\!](\overline{\gamma}_r)(\overline{a})}_{b_{m_i}}) \sqsubseteq [\![\tau_i]\!](\overline{\varphi})(\underbrace{\text{eval}_{\text{P}}[\![\rho'_1]\!]}_{c_1}, \ldots, \underbrace{\text{eval}_{\text{P}}[\![\rho'_{m_i}]\!]}_{c_{m_i}}),$$

  за произволни непрекъснати изображения $\overline{\varphi}$.

  Като обединим всичко от по-горе, получаваме следното:

  $$\begin{aligned} [\![\tau]\!](\overline{\gamma}_r)(\overline{a}) &= [\![\texttt{f}_i(\rho_1, \ldots, \rho_{m_i})]\!](\overline{\gamma}_r)(\overline{a}) \\ &= \gamma_r^i(\underbrace{[\![\rho_1]\!](\overline{\gamma}_r)(\overline{a})}_{b_1}, \ldots, \underbrace{[\![\rho_{m_i}]\!](\overline{\gamma}_r)(\overline{a})}_{b_{m_i}}) && \text{// стойност на терма} \\ &= \gamma_r^i(b_1, \ldots, b_{m_i}) \\ &\sqsubseteq \gamma_r^i(c_1, \ldots, c_{m_i}) && \text{// } \gamma_r^i \text{ е непр. и следователно мон.} \\ &= [\![\tau_i]\!](\overline{\gamma}_{r-1})(c_1, \ldots, c_{m_i}) && \text{// } \gamma_r^i \stackrel{\text{деф}}{=} [\![\tau_i]\!](\overline{\gamma}_{r-1}) \\ &= [\![\tau_i]\!](\overline{\gamma}_{r-1})(\underbrace{\text{eval}_{\text{P}}[\![\rho'_1]\!]}_{c_1}, \ldots, \underbrace{\text{eval}_{\text{P}}[\![\rho'_{m_i}]\!]}_{c_{m_i}}) && \text{// } c_i \stackrel{\text{деф}}{=} \text{eval}_{\text{P}}[\![\rho'_i]\!] \\ &\sqsubseteq \text{eval}_{\text{P}}[\![\tau_i[\texttt{x}_1/\rho'_1, \ldots, \texttt{x}_{m_i}/\rho'_{m_i}]]\!] && \text{// от } \texttt{Include}(r-1) \\ &= \text{eval}_{\text{P}}[\![\texttt{f}_i(\rho'_1, \ldots, \rho'_{m_i})]\!] && \text{// от правило (4)} \\ &= \text{eval}_{\text{P}}[\![\texttt{f}_i(\rho_1[\bar{\texttt{x}}/\overline{\mu}], \ldots, \rho_{m_i}[\bar{\texttt{x}}/\overline{\mu}])]\!] && \text{// } \rho'_j \stackrel{\text{деф}}{=} \rho[\bar{\texttt{x}}/\overline{\mu}] \\ &= \text{eval}_{\text{P}}[\![\texttt{f}_i(\rho_1, \ldots, \rho_{m_i})[\bar{\texttt{x}}/\overline{\mu}]]\!] && \text{// правила за замяна} \\ &= \text{eval}_{\text{P}}[\![\tau[\bar{\texttt{x}}/\overline{\mu}]]\!]. \end{aligned}$$

  Заключаваме, че

  $$[\![\tau]\!](\overline{\gamma}_r)(\underbrace{\text{eval}_{\text{P}}[\![\mu_1]\!]}_{a_1}, \ldots, \underbrace{\text{eval}_{\text{P}}[\![\mu_n]\!]}_{a_n}) \sqsubseteq \text{eval}_{\text{P}}[\![\tau[\bar{\texttt{x}}/\overline{\mu}]]\!].$$

Така доказахме $\mathtt{Include}(r)$.

Най-накрая заключаваме, че $(\forall r)\mathtt{Include}(r)$. $\square$

Следствие 3.2. Нека μ е функционален терм. Тогава $\mathrm{eval}_{\mathtt{P}}[\![\mu]\!]$ е горна граница на веригата $([\![\mu]\!](\overline{\gamma}_r))_{r=0}^{\infty}$.

Лема 3.4. За всяка рекурсивна програма `P`, произволен *функционален* терм μ,

$$[\![\mu]\!](\overline{\gamma}) \sqsubseteq \mathrm{eval}_{\mathtt{P}}[\![\mu]\!],$$

където $\overline{\gamma} = \mathtt{lfp}(\Gamma)$, а $\Gamma = [\![\tau_1]\!] \times [\![\tau_2]\!] \times \cdots \times [\![\tau_k]\!]$ е операторът, който съответства на програмата `P`.

Γ е непрекъснат според *Задача* 1.11.

Упътване.

$$\begin{aligned} [\![\mu]\!](\overline{\gamma}) &= [\![\mu]\!](\bigsqcup_r \overline{\gamma}_r) && // \ \overline{\gamma} = \bigsqcup_r \overline{\gamma}_r \\ &= \bigsqcup_r [\![\mu]\!](\overline{\gamma}_r) && // \text{ от } \textit{Лема } 3.2 \\ &\sqsubseteq \mathrm{eval}_{\mathtt{P}}[\![\mu]\!]. && // \text{ от } \textit{Следствие } 3.2 \end{aligned}$$

$\square$

Теорема 3.2. За всяка рекурсивна програма `P` на езика `FUN` е изпълнено:

$$\mathcal{O}[\![\mathrm{P}]\!] = \mathcal{D}[\![\mathrm{P}]\!].$$

Доказателство. Ние вече знаем от *Следствие* 3.1, че

$$\mathcal{O}[\![\mathrm{P}]\!] \sqsubseteq \mathcal{D}[\![\mathrm{P}]\!].$$

Озн. $\overline{\gamma} = (\gamma_1, \dots, \gamma_n)$ е най-малкото решение на системата от оператори съответстващи на програмата `P`

Остава да докажем обратната посока, а именно

$$\mathcal{D}[\![\mathrm{P}]\!] \sqsubseteq \mathcal{O}[\![\mathrm{P}]\!].$$

За произволни числа $\overline{a}$, имаме следните връзки:

Случаят, когато $\bot$ е измежду $\overline{a}$ е очевиден.

$$\begin{aligned} \mathcal{D}[\![\mathrm{P}]\!](\overline{a}) &= \gamma_1(\overline{a}) && // \text{ деф. на денот. сем.} \\ &= [\![\tau_1]\!](\overline{\gamma})(\overline{a}) && // \ \gamma_1 \stackrel{\text{деф}}{=} [\![\tau_1]\!](\overline{\gamma}) \\ &= [\![\tau_1[\bar{\mathtt{x}}/\bar{\mathtt{a}}]]\!](\overline{\gamma}) && // \text{ Лема за замяната} \end{aligned}$$

$$\begin{aligned}
&\sqsubseteq \mathrm{eval}_{\mathtt{P}}[\![\tau_1[\bar{\mathtt{x}}/\bar{\mathtt{a}}]]\!] && \text{// } \textit{Твърдение } 3.3\\
&= \mathrm{eval}_{\mathtt{P}}[\![\mathtt{f}_1(\mathtt{a}_1, \dots, \mathtt{a}_{m_1})]\!] && \text{// правило (cbn) на опер. сем.}\\
&= \mathcal{O}[\![\mathtt{P}]\!](\overline{a}). && \text{// деф. на опер. сем.}
\end{aligned}$$

□

Глава 4

Езикът PCF

4.1 Синтаксис

PCF- programming language for computable functions. Plotkin's language PCF is often called the *E. coli* of programming languages, the subject of countless studies of language concept.
Тази глава се оповава основно на [8, Глава 19] и [6].

Можем да си мислим за изразите като дървета.

- Типове

$$\mathtt{a} ::= \mathtt{nat} \mid \mathtt{a} \to \mathtt{a}.$$

Когато пишем $\mathtt{a} \to \mathtt{b} \to \mathtt{c}$, то имаме предвид, че $\mathtt{a} \to (\mathtt{b} \to \mathtt{c})$.

- Изрази

$$\begin{aligned}\tau ::= {} & \mathtt{n} \mid \mathtt{x} \mid \tau_1 + \tau_2 \mid \tau_1 - \tau_2 \mid \tau_1 \mathrel{\mathtt{==}} \tau_2 \mid \mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3 \mid \\ & \tau_1\tau_2 \mid \lambda\, \mathtt{x} : \mathtt{a}\, .\, \tau_1 \mid \mathtt{fix}(\tau_1).\end{aligned}$$

Ще означаваме съвкупоността от всички изрази с $\mathcal{E}$, а съвкупността от всички променливи с $\mathcal{V}$. Да обърнем внимание, че не всички изрази са „смислени". Например, $\lambda\, \mathtt{x} : \mathtt{nat}\, .\, \mathtt{xx}$ не е ясно какво означава. След малко ще дефинираме типизираща релация, с чиято помощ ще изхвърлим безслислените изрази.

Понеже тук вече имаме свободни и свързани променливи, трябва да сме внимателни, когато правим замяна на един израз с друг. Да разгледаме функцията $\mathtt{fv} : \mathcal{E} \to \mathcal{P}(\mathcal{V})$, дефинирана със структурна индукция по построението на термовете по следния начин:

- $\mathtt{fv}(\mathtt{n}) = \emptyset$;
- $\mathtt{fv}(\mathtt{x}) = \{\mathtt{x}\}$;
- $\mathtt{fv}(\tau_1 + \tau_2) = \mathtt{fv}(\tau_1 \mathrel{\mathtt{==}} \tau_2) = \mathtt{fv}(\tau_1\tau_2) = \mathtt{fv}(\tau_1) \cup \mathtt{fv}(\tau_2)$;
- $\mathtt{fv}(\mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3) = \mathtt{fv}(\tau_1) \cup \mathtt{fv}(\tau_2) \cup \mathtt{fv}(\tau_3)$;
- $\mathtt{fv}(\lambda\, \mathtt{x} : \mathtt{a}\, .\, \tau) = \mathtt{fv}(\tau) \setminus \{x\}$.
- $\mathtt{fv}(\mathtt{fix}(\tau)) = \mathtt{fv}(\tau)$;

Ще казваме, че един израз τ е **затворен**, ако $\mathtt{fv}(\tau) = \emptyset$. В противен случай, ще казваме, че изразът е **отворен**.

Стойностите понякога се наричат и канонични форми [12].

Определение 4.1. Ще казваме, че един израз $\mathtt{v}$ е **стойност**, ако той е затворен терм, съставен по следния начин:

$$\mathtt{v} ::= \mathtt{n} \mid \lambda\, \mathtt{x} : \mathtt{a}\ .\ \mu,$$

където μ е израз и $\mathtt{fv}(\mu) \subseteq \{\mathtt{x}\}$.

Интуицията тук е, че стойностите са затворени термове, в които не е възможно да се правят повече опростявания. Например, `5 + 6` не е стойност, защото може да се опрости до `11`, но $\lambda\, \mathtt{x} : \mathtt{nat}\ .\ \mathtt{5+6}$ е стойност.

С $\tau\{\mathtt{x}/\rho\}$ ще означаваме изразът получен от τ, в който всяко *свободно* срещане на променливата $\mathtt{x}$ е заменена с израза ρ. Можем да дадем формална дефиниция с индукция по построението на изразите:

Операцията $\{\mathtt{x}/\rho\}$ заменя в дървото за израза τ, всяко листо означено с x, което не участва в дърво с корен $\lambda x : a$, с дървото за израза ρ [7, стр. 35].

- Ако $\tau \equiv \mathtt{n}$, то
$$\tau\{\mathtt{x}/\rho\} \equiv \mathtt{n}.$$
- Ако $\tau \equiv \mathtt{x}$, то
$$\tau\{\mathtt{x}/\rho\} \equiv \rho.$$
- Ако $\tau \equiv \mathtt{y}$, където $\mathtt{y} \not\equiv \mathtt{x}$, то
$$\tau\{\mathtt{x}/\rho\} \equiv \mathtt{y}.$$
- Ако $\tau \equiv \tau_1 + \tau_2$, то
$$\tau\{\mathtt{x}/\rho\} \equiv \tau_1\{\mathtt{x}/\rho\} + \tau_2\{\mathtt{x}/\rho\}.$$
- Ако $\tau \equiv \tau_1 \mathtt{==} \tau_2$, то
$$\tau\{\mathtt{x}/\rho\} \equiv \tau_1\{\mathtt{x}/\rho\} \mathtt{==} \tau_2\{\mathtt{x}/\rho\}.$$
- Ако $\tau \equiv \mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3$, то
$$\tau\{\mathtt{x}/\rho\} \equiv \mathtt{if}\ \tau_1\{\mathtt{x}/\rho\}\ \mathtt{then}\ \tau_2\{\mathtt{x}/\rho\}\ \mathtt{else}\ \tau_3\{\mathtt{x}/\rho\}.$$
- Ако $\tau = \tau_1\tau_2$, то
$$\tau\{\mathtt{x}/\rho\} \equiv \tau_1\{\mathtt{x}/\rho\}\tau_2\{\mathtt{x}/\rho\}.$$
- Ако $\tau = \mathtt{fix}(\tau_1)$, то
$$\tau\{\mathtt{x}/\rho\} \equiv \mathtt{fix}(\tau_1\{\mathtt{x}/\rho\}).$$
- Ако $\tau \equiv \lambda\, \mathtt{x} : \mathtt{a}\ .\ \tau'$, то
$$\tau\{\mathtt{x}/\rho\} \equiv \tau.$$
- Ако $\tau \equiv \lambda\, \mathtt{y} : \mathtt{a}\ .\ \tau'$ и $\mathtt{y} \not\equiv \mathtt{x}$, то
$$\tau\{\mathtt{x}/\rho\} \equiv \lambda\, \mathtt{y} : \mathtt{a}\ .\ \tau'\{\mathtt{x}/\rho\}.$$

Нека $\tau \equiv \lambda\,\mathtt{x} : \mathtt{a}\,.\,\mathtt{x+y}$. Обърнете внимание, че $\tau\{\mathtt{y/x}\} \equiv \lambda\,\mathtt{x} : \mathtt{a}\,.\,\mathtt{x+x}$, т.е. тук получаваме израз, който „смислово” е доста различен от първоначалния израз τ. Проблемът се състои в това, че при замяната на $\mathtt{x}$ с ρ, някоя свободна променлива на ρ може да попадне под обхвата на някоя свързана променлива на τ.

Това е подходът на Хаскел Къри за дефиниране на замяна на променлива с израз [3, стр. 578].

Сега ще дефинираме бинарна релация между изрази, която ще наричаме α-еквивалентност. Интуитивно, всеки два α-еквивалентни израза трябва да бъдат смислово неотличими. Това е най-малката релация между изрази, която ще означаваме с $\equiv_\alpha$, за която са изпълнени свойствата:

В тази дефиниция единствено последният случай е интересен. Например,

$$(\lambda\,\mathtt{x} : \mathtt{a}\,.\,\mathtt{x+z})\mathtt{x} \equiv_\alpha (\lambda\,\mathtt{y} : \mathtt{a}\,.\,\mathtt{y+z})\mathtt{x},$$

но

$$(\lambda\,\mathtt{x} : \mathtt{a}\,.\,\mathtt{x+y})\mathtt{x} \not\equiv_\alpha (\lambda\,\mathtt{y} : \mathtt{a}\,.\,\mathtt{x+y})\mathtt{x}.$$

- $\mathtt{x} \equiv_\alpha \mathtt{x}$;
- $\mathtt{n} \equiv_\alpha \mathtt{n}$;
- ако $\tau_1 \equiv_\alpha \rho_1$ и $\tau_2 \equiv_\alpha \rho_2$, то имаме, че

$$\begin{aligned}&\tau_1 + \tau_2 \equiv_\alpha \rho_1 + \rho_2,\\ &\tau_1 \mathtt{==} \tau_2 \equiv_\alpha \rho_1 \mathtt{==} \rho_2,\\ &\tau_1\tau_2 \equiv_\alpha \rho_1\rho_2;\end{aligned}$$

- ако $\tau_1 \equiv_\alpha \rho_1$, $\tau_2 \equiv_\alpha \rho_2$ и $\tau_3 \equiv_\alpha \rho_3$, то

$$\mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3 \equiv_\alpha \mathtt{if}\ \rho_1\ \mathtt{then}\ \rho_2\ \mathtt{else}\ \rho_3;$$

- ако $\tau_1 \equiv_\alpha \rho_1$ и $\tau_2 \equiv_\alpha \rho_2$, то

$$\tau_1\tau_2 \equiv_\alpha \rho_1\rho_2$$

- ако $\tau \equiv_\alpha \rho$, то

$$\mathtt{fix}(\tau) \equiv_\alpha \mathtt{fix}(\rho);$$

- ако $\tau\{\mathtt{x/z}\} \equiv_\alpha \rho\{\mathtt{y/z}\}$, където $\mathtt{z} \notin \mathtt{var}(\tau) \cup \mathtt{var}(\rho)$, то

$$\lambda\,\mathtt{x} : \mathtt{a}\,.\,\tau \equiv_\alpha \lambda\,\mathtt{y} : \mathtt{a}\,.\,\rho.$$

Например,

$$\lambda\,\mathtt{x} : \mathtt{a}\,.\,\mathtt{x+y} \equiv_\alpha \lambda\,\mathtt{z} : \mathtt{a}\,.\,\mathtt{z+y},$$

но

$$\lambda\,\mathtt{x} : \mathtt{a}\,.\,\mathtt{x+y} \not\equiv_\alpha \lambda\,\mathtt{y} : \mathtt{a}\,.\,\mathtt{y+y},$$

защото

$$(\mathtt{x+y})\{\mathtt{x/u}\} \not\equiv_\alpha (\mathtt{y+y})\{\mathtt{y/u}\}.$$

Също така,

$$\lambda\,\mathtt{x} : \mathtt{nat}\,.\,\mathtt{x}\ \mathtt{+}\ \mathtt{fix}(\lambda\,\mathtt{x} : \mathtt{nat}\,.\,\mathtt{x}) \equiv_\alpha \lambda\,\mathtt{y} : \mathtt{nat}\,.\,\mathtt{y}\ \mathtt{+}\ \mathtt{fix}(\lambda\,\mathtt{z} : \mathtt{nat}\,.\,\mathtt{z}).$$

Определение 4.2. PCF терм е клас на еквивалентност от PCF изрази относно релацията α-еквивалентност.

Това да се премести в нова глава за ламбда смятане.

Де Бройн има по-програмистки подход за дефинирането на PCF термовете. Всеки PCF терм има единствено представяне като израз, в който свързаните променливи са заменени с индекси.

Пример ......

Сега искаме $\tau[\mathtt{x}/\rho]$ да означава изразът получен от израза τ, в който всяко *свободно* срещане на променливата $\mathtt{x}$ е заменена с израза ρ. Тук имаме потенциален проблем. Искаме да направим тази замяна по такъв начин, че свободни променливи на ρ да не попаднат под обхвата на свързани променливи от τ. За да направим това, операцията $[\mathtt{x}/\rho]$ трябва да работи не върху отделни изрази, а върху термове. Можем да дадем формална дефиниция с индукция по построението на термовете:

- Ако $\tau \equiv \mathtt{n}$, то
$$\tau[\mathtt{x}/\rho] \equiv \mathtt{n}.$$
- Ако $\tau \equiv \mathtt{x}$, то
$$\tau[\mathtt{x}/\rho] \equiv \rho.$$
- Ако $\tau \equiv \mathtt{y}$ и $\mathtt{y} \not\equiv \mathtt{x}$, то
$$\tau[\mathtt{x}/\rho] \equiv \mathtt{y}.$$
- Ако $\tau \equiv \tau_1 + \tau_2$, то
$$\tau[\mathtt{x}/\rho] \equiv \tau_1[\mathtt{x}/\rho] + \tau_2[\mathtt{x}/\rho].$$
- Ако $\tau \equiv \tau_1 \mathtt{==} \tau_2$, то
$$\tau[\mathtt{x}/\rho] \equiv \tau_1[\mathtt{x}/\rho] \mathtt{==} \tau_2[\mathtt{x}/\rho].$$
- Ако $\tau \equiv \mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3$, то
$$\tau[\mathtt{x}/\rho] \equiv \mathtt{if}\ \tau_1[\mathtt{x}/\rho]\ \mathtt{then}\ \tau_2[\mathtt{x}/\rho]\ \mathtt{else}\ \tau_3[\mathtt{x}/\rho].$$
- Ако $\tau \equiv \tau_1\tau_2$, то
$$\tau[\mathtt{x}/\rho] \equiv \tau_1[\mathtt{x}/\rho]\tau_2[\mathtt{x}/\rho].$$
- Ако $\tau \equiv \mathtt{fix}(\tau_1)$, то
$$\tau[\mathtt{x}/\rho] \equiv \mathtt{fix}(\tau_1[\mathtt{x}/\rho]).$$
- Ако $\tau \equiv \lambda\ \mathtt{y} : \mathtt{a}\ .\ \tau'$, то
$$\tau[\mathtt{x}/\rho] \equiv \lambda\ \mathtt{z} : \mathtt{a}\ .\ (\tau'[\mathtt{y}/\mathtt{z}][\mathtt{x}/\rho]),$$
  където $\mathtt{z} \not\in \mathtt{fv}(\tau') \cup \mathtt{fv}(\rho) \cup \{\mathtt{x}\}$.

В тази дефиниция отново единствено последният случай е интересен. Обърнете внимание, че според него заместването на $\mathtt{x}$ с ρ дава като резултат безкрайно много изрази, всички от които са α-еквивалентни, т.е. ако работим на ниво термове операцията е добре дефинирана.

Това означава, че няма значение дали ще говорим за $\lambda\ \mathtt{x} : \mathtt{a}\ .\ \mathtt{x+y}$ или за $\lambda\ \mathtt{y} : \mathtt{a}\ .\ \mathtt{y+z}$. Тези два израза описват един и същи терм и тези два израза са α-еквивалентни.

4.2 Добре типизирани термове

Типовите контексти представляват крайни редици от двойки от вида $\mathtt{x} : \mathtt{a}$, т.е.

$$\Gamma ::= \emptyset \mid \Gamma, \mathtt{x} : \mathtt{a}.$$

Например,

$$\Gamma = \mathtt{x} : \mathtt{nat},\ \mathtt{y} : \mathtt{nat} \rightarrow \mathtt{nat},\ \mathtt{z} : \mathtt{nat}.$$

На един типов контекст Γ може да се гледа и като на *крайна* функция приемаща като аргумент променлива и връщаща тип. Например, може понякога да пишем $\Gamma(\mathtt{x}) = \mathtt{nat}$, когато искаме да кажем, че променливата $\mathtt{x}$ има тип $\mathtt{nat}$ в типовия контекст Γ. Ние искаме да работим само с коректно типизирани термове. Например, не е ясно какво означава терма $\mathtt{1\ +}\ \lambda\ \mathtt{x\ :\ nat\ .\ x}$, защото трябва да съберем число и функция - два обекта от различен тип. Ако в изразите имаме свободни променливи, то дали един израз е коректно типизиран ще зависи от типовия контекст. Например, термът $\lambda\,\mathtt{x} : \mathtt{nat}\ .\ \mathtt{z+x}$ е добре типизиран в контекста $\Gamma = \mathtt{z} : \mathtt{nat}$, но не е добре типизиран в контекста $\Delta = \mathtt{z} : \mathtt{nat} \rightarrow \mathtt{nat}$. Сега ще дефинираме релация $\Gamma \vdash \tau : \mathtt{a}$, която ще ни казва, че термът τ, относно типовият контекст Γ, е добре типизиран и има тип $\mathtt{a}$.

Обикновено ще означаваме типовите контексти с главните гръцки $\Gamma, \Delta, \ldots$.

Γ също се нарича и type environment. На англ. $\Gamma \vdash \tau$ се нарича typing judgement.

Обърнете внимание, че можем да правим доказателства свойства на типизиращата релация с индукция по построението на термовете.

$$\frac{}{\Gamma \vdash \mathtt{n} : \mathtt{nat}}\ (\text{const})$$

$$\frac{\mathtt{x} \in \mathtt{dom}(\Gamma) \qquad \Gamma(\mathtt{x}) = \mathtt{a}}{\Gamma \vdash \mathtt{x} : \mathtt{a}}\ (\text{var})$$

$$\frac{\Gamma \vdash \tau_1 : \mathtt{nat} \qquad \Gamma \vdash \tau_2 : \mathtt{nat}}{\Gamma \vdash \tau_1 + \tau_2 : \mathtt{nat}}\ (\text{plus})$$

$$\frac{\Gamma \vdash \tau_1 : \mathtt{nat} \qquad \Gamma \vdash \tau_2 : \mathtt{nat}}{\Gamma \vdash \tau_1 - \tau_2 : \mathtt{nat}}\ (\text{minus})$$

$$\frac{\Gamma \vdash \tau_1 : \mathtt{nat} \qquad \Gamma \vdash \tau_2 : \mathtt{nat}}{\Gamma \vdash \tau_1 \mathtt{==} \tau_2 : \mathtt{nat}}\ (\text{eq})$$

$$\frac{\Gamma \vdash \tau_1 : \mathtt{nat} \qquad \Gamma \vdash \tau_2 : \mathtt{a} \qquad \Gamma \vdash \tau_3 : \mathtt{a}}{\Gamma \vdash \mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3 : \mathtt{a}}\ (\text{if})$$

$$\frac{\Gamma \vdash \tau_1 : \mathtt{a} \to \mathtt{b} \qquad \Gamma \vdash \tau_2 : \mathtt{a}}{\Gamma \vdash \tau_1\tau_2 : \mathtt{b}}\ (\text{app})$$

$$\frac{\Gamma \vdash \tau : \mathtt{a} \to \mathtt{a}}{\Gamma \vdash \mathtt{fix}(\tau) : \mathtt{a}}\ (\text{fix})$$

$$\frac{\mathtt{x} \notin \mathtt{dom}(\Gamma) \qquad \Gamma, \mathtt{x} : \mathtt{a} \vdash \tau : \mathtt{b}}{\Gamma \vdash \lambda \mathtt{x} : \mathtt{a} \,.\, \tau : \mathtt{a} \to \mathtt{b}}\ (\text{lambda})$$

Фигура 4.1: Релация за типизиране на термовете от езика PCF

Ако имаме затворен израз τ, то ще пишем $\tau : \mathtt{a}$ вместо $\emptyset \vdash \tau : \mathtt{a}$. Да положим

$$\mathtt{PCF}_{\mathtt{a}} \stackrel{\text{деф}}{=} \{\tau \text{ е затворен терм } \mid \emptyset \vdash \tau : \mathtt{a}\}$$

$$\mathtt{Value}_{\mathtt{a}} \stackrel{\text{деф}}{=} \{\tau \text{ е стойност } \mid \emptyset \vdash \tau : \mathtt{a}\}.$$

Твърдение 4.1 (Weakening Lemma)**.** За произволен типов контекст Γ и терм τ, можем да направим извода:

$$\frac{\Gamma \vdash \tau : \mathtt{b} \qquad \mathtt{x} \notin \mathtt{dom}(\Gamma)}{\Gamma, \mathtt{x} : \mathtt{a} \vdash \tau : \mathtt{b}}$$

Твърдение 4.2 (Exchange Lemma)**.** За произволни типови контексти Γ и Γ', терм τ и променливи $\mathtt{x}$ и $\mathtt{y}$, можем да направим извода:

$$\frac{\Gamma, \mathtt{x} : \mathtt{a}, \mathtt{y} : \mathtt{b}, \Gamma' \vdash \tau : \mathtt{c}}{\Gamma, \mathtt{y} : \mathtt{b}, \mathtt{x} : \mathtt{a}, \Gamma' \vdash \tau : \mathtt{c}}$$

Твърдение 4.3. Ако $\Gamma \vdash \tau : \mathtt{a}$, то $\mathtt{fv}(\tau) \subseteq \mathtt{dom}(\Gamma)$.

Твърдение 4.4. Ако $\Gamma \vdash \tau : \mathtt{a}$ и $\Gamma \vdash \tau : \mathtt{b}$, то $\mathtt{a} = \mathtt{b}$.

Следствие 4.1. Всеки затворен терм има най-много един тип.

Задача 4.1. Докажете или опровергайте дали е възможно да съществува типов контекст Γ и тип $\mathtt{a}$, такива че $\Gamma \vdash \mathtt{xx} : \mathtt{a}$. [10, стр. 104]

Лема 4.1. Типизиращата релация е съвместима с операцията субституция за термове. С други думи, имаме извода:

$$\frac{\Gamma \vdash \rho : \mathtt{a} \qquad \mathtt{x} \notin \mathtt{dom}(\Gamma) \qquad \Gamma, \mathtt{x} : \mathtt{a} \vdash \tau : \mathtt{b}}{\Gamma \vdash \tau[\mathtt{x}/\rho] : \mathtt{b}}$$

Доказателство. Индукция по построението на термовете. Започваме с базовият случай, който може да се разбие на три подслучая.

- Нека $\tau \equiv \mathtt{n}$. Тогава е ясно, че $\mathtt{b} = \mathtt{nat}$ и $\mathtt{n}[\mathtt{x}/\rho] \equiv \mathtt{n}$. Оттук веднага получаваме, че
$$\Gamma \vdash \tau[\mathtt{x}/\rho] : \mathtt{b}.$$
- Нека $\tau \equiv \mathtt{x}_i$, където $\mathtt{x}_i \not\equiv \mathtt{x}$. Тогава, щом $\Gamma, \mathtt{x} : \mathtt{a} \vdash \mathtt{x}_i : \mathtt{a}_i$, то $\mathtt{b} = \mathtt{a}_i$. Тук е ясно, че $\mathtt{x}_i[\mathtt{x}/\rho] \equiv \mathtt{x}_i$ и тогава веднага получаваме, че
$$\Gamma \vdash \tau[\mathtt{x}/\rho] : \mathtt{b}.$$
- Нека $\tau \equiv \mathtt{x}$. Тогава, щом $\Gamma, \mathtt{x} : \mathtt{a} \vdash \mathtt{x} : \mathtt{b}$, то $\mathtt{b} = \mathtt{a}$. Тук е ясно, че $\mathtt{x}[\mathtt{x}/\rho] \equiv \rho$ и тогава веднага получаваме, че
$$\Gamma \vdash \tau[\mathtt{x}/\rho] : \mathtt{b}.$$

Сега преминаваме към индукционната стъпка.

- Нека $\tau \equiv \tau_1 + \tau_2$. За първата част, щом $\Gamma, \mathtt{x} : \mathtt{a} \vdash \tau_1 + \tau_2 : \mathtt{b}$, то е ясно, че $\mathtt{b} = \mathtt{nat}$. Имаме, че
$$\frac{\Gamma, \mathtt{x} : \mathtt{a} \vdash \tau_1 : \mathtt{nat} \qquad \Gamma, \mathtt{x} : \mathtt{a} \vdash \tau_2 : \mathtt{nat}}{\Gamma, \mathtt{x} : \mathtt{a} \vdash \tau_1 + \tau_2 : \mathtt{nat}}\ \text{(plus)}$$

  Сега можем да приложим **(И.П.)** и получаваме следния извод:

$$\dfrac{\dfrac{\dfrac{\text{(от условието)}}{\Gamma \vdash \rho : \mathtt{a}} \qquad \Gamma, \mathtt{x} : \mathtt{a} \vdash \tau_1 : \mathtt{nat}}{\Gamma \vdash \tau_1[\mathtt{x}/\rho] : \mathtt{nat}}\ \textbf{(И.П.)} \qquad \dfrac{\dfrac{\text{(от условието)}}{\Gamma \vdash \rho : \mathtt{a}} \qquad \Gamma, \mathtt{x} : \mathtt{a} \vdash \tau_2 : \mathtt{nat}}{\Gamma \vdash \tau_2[\mathtt{x}/\rho] : \mathtt{nat}}\ \textbf{(И.П.)}}{\dfrac{\Gamma \vdash \tau_1[\mathtt{x}/\rho] + \tau_2[\mathtt{x}/\rho] : \mathtt{nat}}{\Gamma \vdash \tau[\mathtt{x}/\rho] : \mathtt{nat}}\ \text{(правила за замяна)}}\ \text{(plus)}$$

- Нека $\tau \equiv \tau_1 - \tau_2$.
- Нека $\tau \equiv \tau_1 \texttt{==} \tau_2$.
- Нека $\tau \equiv \texttt{if}\ \tau_1\ \texttt{then}\ \tau_2\ \texttt{else}\ \tau_3$.
- Нека $\tau \equiv \tau_1\tau_2$. Понеже имаме, че

$$\dfrac{\Gamma, \mathtt{x} : \mathtt{a} \vdash \tau_1 : \mathtt{c} \to \mathtt{b} \qquad \Gamma, \mathtt{x} : \mathtt{a} \vdash \tau_2 : \mathtt{c}}{\Gamma, \mathtt{x} : \mathtt{a} \vdash \tau_1\tau_2 : \mathtt{b}}\ \text{(app)}$$

то можем да приложим **(И.П.)** за да получим, че

$$\dfrac{\dfrac{\overline{\Gamma \vdash \rho : \mathtt{a}}^{\text{от условието}} \qquad \Gamma, \mathtt{x} : \mathtt{a} \vdash \tau_1 : \mathtt{c} \to \mathtt{b}}{\Gamma \vdash \tau_1[\mathtt{x}/\rho] : \mathtt{b}}\ \textbf{(И.П.)} \qquad \dfrac{\overline{\Gamma \vdash \rho : \mathtt{a}}^{\text{от условието}} \qquad \Gamma, \mathtt{x} : \mathtt{a} \vdash \tau_2 : \mathtt{c} \to \mathtt{b}}{\Gamma \vdash \tau_2[\mathtt{x}/\rho] : \mathtt{c}}\ \textbf{(И.П.)}}{\dfrac{\Gamma \vdash \tau_1[\mathtt{x}/\rho](\tau_2[\mathtt{x}/\rho]) : \mathtt{c}}{\Gamma \vdash \tau[\mathtt{x}/\rho] : \mathtt{c}}\ \text{(правила на замяна)}}\ \text{(app)}$$

- Нека $\tau \equiv \mathtt{fix}(\tau')$. От правилата за типизиране е ясно, че имаме следния извод:

$$\dfrac{\Gamma, \mathtt{x} : \mathtt{a} \vdash \tau' : \mathtt{b} \to \mathtt{b}}{\Gamma, \mathtt{x} : \mathtt{a} \vdash \mathtt{fix}(\tau') : \mathtt{b}}\ \text{(fix)}$$

Сега можем да приложим **(И.П.)** за терма τ'. Получаваме, че

$$\dfrac{\dfrac{\dfrac{\overline{\Gamma \vdash \rho : \mathtt{a}}^{\text{от условието}} \qquad \Gamma, \mathtt{x} : \mathtt{a} \vdash \tau' : \mathtt{b} \to \mathtt{b}}{\Gamma \vdash \tau'[\mathtt{x}/\rho] : \mathtt{b} \to \mathtt{b}}\ \textbf{(И.П.)}}{\Gamma \vdash \mathtt{fix}(\tau'[\mathtt{x}/\rho]) : \mathtt{b}}\ \text{(fix)}}{\Gamma \vdash \mathtt{fix}(\tau')[\mathtt{x}/\rho] : \mathtt{b}}\ \text{(правила за замяна)}$$

- Нека $\tau \equiv \lambda\, \mathtt{y} : \mathtt{c}\, .\, \tau'$, където $\mathtt{y} \notin \mathtt{dom}(\Gamma) \cup \{\mathtt{x}\}$.

  От правилата за типизиране е ясно, че щом $\Gamma, \mathtt{x} : \mathtt{a} \vdash \tau : \mathtt{b}$, то типът $\mathtt{b}$ е такъв, че $\mathtt{b} = \mathtt{c} \to \mathtt{d}$, за някой тип $\mathtt{d}$, и имаме извода:

$$\dfrac{\mathtt{y} \notin \mathtt{dom}(\Gamma) \cup \{\mathtt{x}\} \qquad \Gamma, \mathtt{x} : \mathtt{a}, \mathtt{y} : \mathtt{c} \vdash \tau' : \mathtt{d}}{\Gamma, \mathtt{x} : \mathtt{a} \vdash \lambda\, \mathtt{y} : \mathtt{c}\, .\, \tau' : \mathtt{c} \to \mathtt{d}}\ \text{(lambda)}$$

  Това означава, че можем да използваме **(И.П.)** за терма τ' и така получаваме, че

$$\dfrac{\dfrac{\mathtt{y} \notin \mathtt{dom}(\Gamma) \qquad \dfrac{\dfrac{\mathtt{y} \notin \mathtt{dom}(\Gamma) \qquad \overline{\Gamma \vdash \rho : \mathtt{a}}^{\text{от условието}}}{\Gamma, \mathtt{y} : \mathtt{c} \vdash \rho : \mathtt{a}} \qquad \dfrac{\Gamma, \mathtt{x} : \mathtt{a}, \mathtt{y} : \mathtt{c} \vdash \tau' : \mathtt{d}}{\Gamma, \mathtt{y} : \mathtt{c}, \mathtt{x} : \mathtt{a} \vdash \tau' : \mathtt{d}}}{\Gamma, \mathtt{y} : \mathtt{c} \vdash \tau'[\mathtt{x}/\rho] : \mathtt{d}}\ \textbf{(И.П.)}}{\Gamma \vdash \lambda\, \mathtt{y} : \mathtt{c}\, .\, \tau'[\mathtt{x}/\rho] : \mathtt{c} \to \mathtt{d}}\ \text{(lambda)}}{\Gamma \vdash (\lambda\, \mathtt{y} : \mathtt{c}\, .\, \tau')[\mathtt{x}/\rho] : \mathtt{c} \to \mathtt{d}}\ \text{(правила за замяна)}$$

$\square$

4.3 Операционна семантика

За затворен терм $\tau : \mathtt{a}$ и стойност $\mathtt{v} : \mathtt{a}$, дефинираме релацията $\tau \Downarrow_{\mathtt{a}}^{\ell} \mathtt{v}$ по следния начин.

Да напомним, че с $\mathtt{v}$ означаваме стойности, които биват или константи или затворени термове от вида $\lambda\, \mathtt{x} : \mathtt{a}\, .\, \mu$. Обърнете внимание, че заради случите (fix) и (cbn) не можем да доказваме свойства на операционната семантика с индукция по построението на термовете. Можем да направим това с индукция по броя на стъпки в изчислението. [7, стр. 109]

$$\frac{\mathtt{v} : \mathtt{a}}{\mathtt{v} \Downarrow_{\mathtt{a}}^{0} \mathtt{v}}\ (\text{val})$$

$$\frac{\tau_1 \Downarrow_{\mathtt{nat}}^{\ell_1} \mathtt{n}_1 \qquad \tau_2 \Downarrow_{\mathtt{nat}}^{\ell_2} \mathtt{n}_2 \qquad n = (\mathtt{==})(n_1, n_2)}{\tau_1 \mathbin{\mathtt{==}} \tau_2 \Downarrow_{\mathtt{nat}}^{\ell_1+\ell_2+1} \mathtt{n}}\ (\text{eq})$$

$$\frac{\tau_1 \Downarrow_{\mathtt{nat}}^{\ell_1} \mathtt{n}_1 \qquad \tau_2 \Downarrow_{\mathtt{nat}}^{\ell_2} \mathtt{n}_2 \qquad n = (\mathtt{+})(n_1, n_2)}{\tau_1 \mathbin{\mathtt{+}} \tau_2 \Downarrow_{\mathtt{nat}}^{\ell_1+\ell_2+1} \mathtt{n}}\ (\text{plus})$$

$$\frac{\tau_1 \Downarrow_{\mathtt{nat}}^{\ell_1} \mathtt{n}_1 \qquad \tau_2 \Downarrow_{\mathtt{nat}}^{\ell_2} \mathtt{n}_2 \qquad n = (\mathtt{-})(n_1, n_2)}{\tau_1 \mathbin{\mathtt{-}} \tau_2 \Downarrow_{\mathtt{nat}}^{\ell_1+\ell_2+1} \mathtt{n}}\ (\text{minus})$$

$$\frac{\tau_1 \Downarrow_{\mathtt{nat}}^{\ell_1} \mathtt{0} \qquad \tau_3 \Downarrow_{\mathtt{a}}^{\ell_2} \mathtt{v}}{\mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3 \Downarrow_{\mathtt{a}}^{\ell_1+\ell_2+1} \mathtt{v}}\ (\text{if}_0)$$

$$\frac{\tau_1 \Downarrow_{\mathtt{nat}}^{\ell_1} \mathtt{n} \qquad \tau_2 \Downarrow_{\mathtt{a}}^{\ell_2} \mathtt{v} \qquad \mathtt{n} \not\equiv \mathtt{0}}{\mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3 \Downarrow_{\mathtt{a}}^{\ell_1+\ell_2+1} \mathtt{v}}\ (\text{if}^{+})$$

$$\frac{\tau_1 \Downarrow_{\mathtt{a}\to\mathtt{b}}^{\ell_1} \lambda\, \mathtt{x} : \mathtt{a}\, .\, \tau_1' \qquad \tau_1'[\mathtt{x}/\tau_2] \Downarrow_{\mathtt{b}}^{\ell_2} \mathtt{v}}{\tau_1 \tau_2 \Downarrow_{\mathtt{b}}^{\ell_1+\ell_2+1} \mathtt{v}}\ (\text{cbn})$$

$$\frac{\tau\ \mathtt{fix}(\tau) \Downarrow_{\mathtt{a}}^{\ell} \mathtt{v}}{\mathtt{fix}(\tau) \Downarrow_{\mathtt{a}}^{\ell+1} \mathtt{v}}\ (\text{fix})$$

Фигура 4.2: Правила на операционната семантика за езика PCF

- Ще пишем $\tau \Downarrow_{\mathtt{a}} \mathtt{v}$, ако съществува ℓ, за което $\tau \Downarrow_{\mathtt{a}}^{\ell} \mathtt{v}$.
- Също така, ще пишем $\tau \not\Downarrow_{\mathtt{a}}$, ако не съществува стойност $\mathtt{v}$, за която $\tau \Downarrow_{\mathtt{a}} \mathtt{v}$.

Лема 4.2. За произволен затворен терм τ и стойности $\mathtt{v}$ и $\mathtt{u}$,

$$\tau \Downarrow_{\mathtt{a}} \mathtt{v} \,\&\, \tau \Downarrow_{\mathtt{a}} \mathtt{u} \implies \mathtt{v} \equiv \mathtt{u}.$$

Доказателство. Индукция по дължината на извода. $\square$

Определение 4.3. За всеки два затворени терма τ_1 и τ_2 от тип $\mathtt{a}$, дефинираме релацията $\tau_1 \leq_{op} \tau_2 : \mathtt{a}$ по следния начин:

- $\emptyset \vdash \tau_1 : \mathtt{a}$ и $\emptyset \vdash \tau_2 : \mathtt{a}$,
- $(\forall \mathtt{v} \in \mathtt{Value}_{\mathtt{a}})[\tau_1 \Downarrow_{\mathtt{a}} \mathtt{v} \implies \tau_2 \Downarrow_{\mathtt{a}} \mathtt{v}]$.

Лесно се съобразява, че релацията $\leq_{op}$ е рефлексивна и транзитивна. Тя поражда релация на еквивалентност $\cong_{op}$ като дефинираме $\tau_1 \cong_{op} \tau_2 : \mathtt{a}$ точно тогава, когато $\tau_1 \leq_{op} \tau_2 : \mathtt{a}$ и $\tau_2 \leq_{op} \tau_1 : \mathtt{a}$.

Пример 4.1. Нека $\mathtt{a} = \mathtt{nat} \to (\mathtt{nat} \to \mathtt{nat})$ и

$$\tau \equiv \mathtt{fix(}\underbrace{\lambda\,\mathtt{f} : \mathtt{a}\,.\,\overbrace{\lambda\,\mathtt{x} : \mathtt{nat}\,.\,\lambda\,\mathtt{y} : \mathtt{nat}\,.\,\mathtt{if}\ \mathtt{y}\ \mathtt{==}\ \mathtt{0}\ \mathtt{then}\ \mathtt{0}\ \mathtt{else}\ \mathtt{x}\ \mathtt{+}\ \mathtt{(f\ x\ (y-1))}}^{\tau_0'}}_{\tau_0}\mathtt{)}.$$

Както се вижда от *Фигура* 4.3, имаме

$$\emptyset \vdash \tau : \mathtt{nat} \to (\mathtt{nat} \to \mathtt{nat}).$$

По-долу ще видим, че

$$\tau\ \mathtt{2}\ \mathtt{3} \Downarrow_{\mathtt{nat}} \mathtt{6}.$$

По-общо, би трябвало да е ясно, че можем да докажем, че за всеки две естествени числа n и k, ако $m = n * k$, то

$$\tau\ \mathtt{n}\ \mathtt{k} \Downarrow_{\mathtt{nat}} \mathtt{m}.$$

Нека сега

$$\rho \equiv \lambda\,\mathtt{g} : \mathtt{a}\,.\,\mathtt{fix}(\lambda\mathtt{f} : \mathtt{nat} \to \mathtt{nat}\,.\,\lambda\,\mathtt{x} : \mathtt{nat}\,.\,\mathtt{if}\ \mathtt{x}\ \mathtt{==}\ \mathtt{0}\ \mathtt{then}\ \mathtt{1}\ \mathtt{else}\ \mathtt{g}\ \mathtt{x}\ \mathtt{f(x-1)}).$$

Лесно се вижда, че

$$\emptyset \vdash \rho : \mathtt{a} \to (\mathtt{nat} \to \mathtt{nat}).$$

Също така, за всяко n, ако $k = n!$, то

$$\rho\,\tau\,\mathtt{n} \Downarrow_{\mathtt{nat}} \mathtt{k}.$$

Можем да напишем директно горния пример и на хаскел:

```
ghci> fix f = f (fix f)
ghci> times = fix(\f x y -> if y == 0 then 0 else x + (f x (y-1)))
ghci> times 2 3
6
ghci> rho = \g -> fix(\f x -> if x == 0 then 1 else g x (f (x-1)))
ghci> fact = rho times
ghci> fact 5
120
```

$$
\dfrac{
  \dfrac{
    \dfrac{\Gamma(\texttt{y}) = \texttt{nat}}{\Gamma \vdash \texttt{y} : \texttt{nat}}
    \quad
    \dfrac{}{\Gamma \vdash \texttt{0} : \texttt{nat}}
  }{\Gamma \vdash \texttt{y==0} : \texttt{nat}}
  \quad
  \dfrac{}{\Gamma \vdash \texttt{0} : \texttt{nat}}
  \quad
  \dfrac{
    \dfrac{\Gamma(\texttt{x}) = \texttt{nat}}{\Gamma \vdash \texttt{x} : \texttt{nat}}
    \quad
    \dfrac{
      \dfrac{
        \dfrac{\Gamma(\texttt{f}) = \texttt{a}}{\Gamma \vdash \texttt{f} : \texttt{a}}
        \quad
        \dfrac{\Gamma(\texttt{x}) = \texttt{nat}}{\Gamma \vdash \texttt{x} : \texttt{nat}}
      }{\Gamma \vdash \texttt{f x} : \texttt{nat} \to \texttt{nat}}
      \quad
      \dfrac{
        \dfrac{\Gamma(\texttt{y}) = \texttt{nat}}{\Gamma \vdash \texttt{y} : \texttt{nat}}
        \quad
        \dfrac{}{\Gamma \vdash \texttt{1} : \texttt{nat}}
      }{\Gamma \vdash \texttt{y-1} : \texttt{nat}}
    }{\Gamma \vdash \texttt{f x (y-1)} : \texttt{nat}}
  }{\Gamma \vdash \texttt{x + f x (y-1)} : \texttt{nat}}
}{
  \dfrac{
    \underbrace{\texttt{f} : \texttt{a}, \texttt{x} : \texttt{nat}, \texttt{y} : \texttt{nat}}_{\Gamma} \vdash \texttt{if y == 0 then 0 else x + (f x (y-1))} : \texttt{nat}
  }{
    \dfrac{
      \texttt{f} : \texttt{a}, \texttt{x} : \texttt{nat} \vdash \lambda\, \texttt{y} : \texttt{nat} \,.\, \texttt{if y == 0 then 0 else x + (f x (y-1))} : \texttt{nat} \to \texttt{nat}
    }{
      \dfrac{\texttt{f} : \texttt{a} \vdash \tau_0' : \texttt{a}}{
        \dfrac{\emptyset \vdash \tau_0 : \texttt{a} \to \texttt{a}}{\emptyset \vdash \underbrace{\texttt{fix}(\tau_0)}_{\tau} : \texttt{a}}
      }
    }
  }
}
$$

Фигура 4.3: Формален извод според правилата на типизиращата релацията, който показва, че $\emptyset \vdash \tau : \texttt{a}$.

След като видяхме, че τ е терм от тип $\texttt{a}$, нека да видим колко стъпки ще са ни нужни, според правилата на операционната семантика, за да проверим, че $\tau\ 3\ 2 \Downarrow_{\texttt{nat}} 6$.

Първо да обърнем внимание, че τ_0 е стойност, т.е. $\tau_0 \Downarrow^0_{\texttt{a}\to\texttt{a}} \tau_0$. Макар и τ_0' да има свободна променлива $\texttt{f}$, понеже термът $\texttt{fix}(\tau_0)$ е затворен с тип $\texttt{a}$, то затвореният терм

$$\tau_0'[\texttt{f}/\texttt{fix}(\tau_0)] \equiv \lambda\, \texttt{x} : \texttt{nat} \,.\, \lambda\, \texttt{y} : \texttt{nat} \,.\, \texttt{if y == 0 then 0 else x + (fix}(\tau_0)\texttt{ x (y-1))}$$

е стойност и следователно, $\tau_0'[\texttt{f}/\texttt{fix}(\tau_0)] \Downarrow^0_{\texttt{a}} \tau_0'[\texttt{f}/\texttt{fix}(\tau_0)]$.

$$
\text{(app)}\;\dfrac{
  \text{(app)}\;\dfrac{
    \text{(fix)}\;\dfrac{
      \text{(app)}\;\dfrac{
        \text{(val)}\;\dfrac{\tau_0 \text{ е стойност}}{\tau_0 \Downarrow^0_{\mathtt{a}\to\mathtt{a}} \lambda\, \mathtt{f} : \mathtt{a}\,.\, \tau_0'}
        \qquad
        \dfrac{\tau_0'[\mathtt{f}/\mathtt{fix}(\tau_0)] \text{ е стойност}}{\tau_0'[\mathtt{f}/\mathtt{fix}(\tau_0)] \Downarrow^0_{\mathtt{a}} \lambda\, \mathtt{x} : \mathtt{nat}\,.\, \tau_2}\;\text{(val)}
      }{\tau_0 \mathtt{fix}(\tau_0) \Downarrow^1_{\mathtt{a}} \lambda\, \mathtt{x} : \mathtt{nat}\,.\, \tau_2}
    }{\mathtt{fix}(\tau_0) \Downarrow^2_{\mathtt{a}} \lambda\, \mathtt{x} : \mathtt{nat}\,.\, \tau_2}
    \qquad
    \text{(val)}\;\dfrac{\tau_2[\mathtt{x}/3] \text{ е стойност}}{\tau_2[\mathtt{x}/3] \Downarrow^0_{\mathtt{nat}\to\mathtt{nat}} \lambda\, \mathtt{y} : \mathtt{nat}\,.\, \tau_1}
  }{\mathtt{fix}(\tau_0)\ 3 \Downarrow^3_{\mathtt{nat}\to\mathtt{nat}} \lambda\, \mathtt{y} : \mathtt{nat}\,.\, \tau_1}
  \qquad
  \dfrac{\textit{Фигура } 4.5}{\tau_1[\mathtt{y}/2] \Downarrow^{16}_{\mathtt{nat}} 6}
}{\underbrace{\mathtt{fix}(\tau_0)}_{\tau}\ 3\ 2 \Downarrow^{20}_{\mathtt{nat}} 6}
$$

Фигура 4.4: Първа част от изчислението на τ 3 2 според правилата на операционната семантика.

Нека за улеснение да положим

$$
\begin{aligned}
\tau_2 &\equiv \lambda\, \mathtt{y} : \mathtt{nat}\,.\, \mathtt{if\ y\ ==\ 0\ then\ 0\ else\ \ x\ +\ (fix}(\tau_0)\mathtt{\ x\ (y-1))} \\
\tau_1 &\equiv \mathtt{if\ y\ ==\ 0\ then\ 0\ else\ \ 3\ +\ (fix}(\tau_0)\mathtt{\ 3\ (y-1))}.
\end{aligned}
$$

Термът τ_2 не е стойност, защото има свободна променлива $\mathtt{x}$, но вече термът $\tau_2[\mathtt{x}/3]$ е стойност. Лесно се съобразява, че

$$
\begin{aligned}
\tau_0'[\mathtt{f}/\mathtt{fix}(\tau_0)] &\equiv \lambda\, \mathtt{x} : \mathtt{nat}\,.\, \tau_2 \\
\tau_2[\mathtt{x}/3] &\equiv \lambda\, \mathtt{y} : \mathtt{nat}\,.\, \tau_1.
\end{aligned}
$$

Сега във *Фигура* 4.5 продължаваме десния клон на изчислението:

$$
\dfrac{
\dfrac{
\dfrac{2 \text{ е стойност}}{\texttt{2} \Downarrow^{0}_{\texttt{nat}} \texttt{2}}\,(\text{val})
\quad
\dfrac{0 \text{ е стойност}}{\texttt{0} \Downarrow^{0}_{\texttt{nat}} \texttt{0}}\,(\text{val})
}{\texttt{2 == 0} \Downarrow^{1}_{\texttt{nat}} \texttt{0}}\,(\text{eq})
\quad
\dfrac{
\dfrac{3 \text{ е стойност}}{\texttt{3} \Downarrow^{0}_{\texttt{nat}} \texttt{3}}
\quad
\dfrac{
\dfrac{\text{(Повтаря се както по-горе)}}{\texttt{fix}(\tau_0)\ \texttt{3} \Downarrow^{3}_{\texttt{nat}\to\texttt{nat}} \lambda\, \texttt{y} : \texttt{nat} \,.\, \tau_1}
\quad
\dfrac{\textit{Фигура } 4.6}{\tau_1[\texttt{y}/\texttt{2-1}] \Downarrow^{10}_{\texttt{nat}} \texttt{3}}
}{\texttt{fix}(\tau_0)\ \texttt{3 (2-1)} \Downarrow^{14}_{\texttt{nat}} \texttt{3}}\,(\text{app})
}{\texttt{3 + (fix}(\tau_0)\texttt{ 3 (2-1))} \Downarrow^{15}_{\texttt{nat}} \texttt{6}}\,(\text{plus})
}{\underbrace{\texttt{if 2 == 0 then 0 else 3 + (fix}(\tau_0)\texttt{ 3 (2-1))}}_{\tau_1[\texttt{y}/\texttt{2}]} \Downarrow^{16}_{\texttt{nat}} \texttt{6}}\,(\text{if}_0)
$$

Фигура 4.5: Втора част от изчислението, която показва, че $\tau_1[\texttt{y}/\texttt{2}] \Downarrow^{16}_{\texttt{nat}} \texttt{6}$.

Във *Фигура* 4.6 продължаваме с десния клон на изчислението, като тук вече имаме, че:

$$\tau_1[\texttt{y}/\texttt{2-1}] \equiv \texttt{if 2-1 == 0 then 0 else 3 + (fix}(\tau_0)\texttt{ 3 (2-1-1))}$$

Най-накрая, във *Фигура* 4.7 завършваме изчислението като използваме, че

$$\tau_1[\texttt{y}/\texttt{2-1-1}] \equiv \texttt{if 2-1-1 == 0 then 0 else 3 + (fix}(\tau_0)\texttt{ 3 (2-1-1-1))}.$$

$$
\text{(if}_0\text{)}\;\dfrac{
\text{(eq)}\;\dfrac{
\text{(minus)}\;\dfrac{\dfrac{\mathtt{2}\text{ е стойност}}{\mathtt{2} \Downarrow^{0}_{\mathtt{nat}} \mathtt{2}} \quad \dfrac{\mathtt{1}\text{ е стойност}}{\mathtt{1} \Downarrow^{0}_{\mathtt{nat}} \mathtt{1}}}{\mathtt{2\text{-}1} \Downarrow^{1}_{\mathtt{nat}} \mathtt{1}}
\quad
\dfrac{\mathtt{0}\text{ е стойност}}{\mathtt{0} \Downarrow^{0}_{\mathtt{nat}} \mathtt{0}}
}{\mathtt{2\text{-}1\ ==\ 0} \Downarrow^{2}_{\mathtt{nat}} \mathtt{0}}
\quad
\dfrac{
\dfrac{\mathtt{3}\text{ е стойност}}{\mathtt{3} \Downarrow^{0}_{\mathtt{nat}} \mathtt{3}}
\quad
\dfrac{
\dfrac{\text{(Повтаря се както по-горе)}}{\mathtt{fix}(\tau_0)\ \mathtt{3} \Downarrow^{3}_{\mathtt{nat}\to\mathtt{nat}} \lambda\,\mathtt{y} : \mathtt{nat}\,.\,\tau_1}
\quad
\dfrac{\textit{Фигура } 4.7}{\tau_1[\mathtt{y}/\mathtt{2\text{-}1\text{-}1}] \Downarrow^{4}_{\mathtt{nat}} \mathtt{0}}
}{\mathtt{fix}(\tau_0)\ \mathtt{3}\ \mathtt{(2\text{-}1\text{-}1)} \Downarrow^{8}_{\mathtt{nat}} \mathtt{0}}\;\text{(app)}
}{\mathtt{3\ +\ (fix}(\tau_0)\ \mathtt{3\ (2\text{-}1\text{-}1))} \Downarrow^{9}_{\mathtt{nat}} \mathtt{3}}\;\text{(plus)}
}{\underbrace{\mathtt{if}\ \mathtt{2\text{-}1\ ==\ 0}\ \mathtt{then}\ \mathtt{0}\ \mathtt{else}\ \ \mathtt{3\ +\ (fix}(\tau_0)\ \mathtt{3\ (2\text{-}1\text{-}1))}}_{\tau_1[\mathtt{y}/\mathtt{2\text{-}1}]} \Downarrow^{10}_{\mathtt{nat}} \mathtt{3}}
$$

Фигура 4.6: Трета част от изчислението, която показва, че $\tau_1[\mathtt{y}/\mathtt{2\text{-}1}] \Downarrow^{10}_{\mathtt{nat}} \mathtt{3}$

$$
\text{(if}^{+}\text{)}\;\dfrac{
\text{(eq)}\;\dfrac{
\text{(minus)}\;\dfrac{
\text{(minus)}\;\dfrac{\dfrac{\mathtt{2}\text{ е стойност}}{\mathtt{2} \Downarrow^{0}_{\mathtt{nat}} \mathtt{2}} \quad \dfrac{\mathtt{1}\text{ е стойност}}{\mathtt{1} \Downarrow^{0}_{\mathtt{nat}} \mathtt{1}}}{\mathtt{2\text{-}1} \Downarrow^{1}_{\mathtt{nat}} \mathtt{1}}
\quad
\dfrac{\mathtt{1}\text{ е стойност}}{\mathtt{1} \Downarrow^{0}_{\mathtt{nat}} \mathtt{1}}
}{\mathtt{2\text{-}1\text{-}1} \Downarrow^{2}_{\mathtt{nat}} \mathtt{0}}
\quad
\dfrac{\mathtt{0}\text{ е стойност}}{\mathtt{0} \Downarrow^{0}_{\mathtt{nat}} \mathtt{0}}
}{\mathtt{2\text{-}1\text{-}1\ ==\ 0} \Downarrow^{3}_{\mathtt{nat}} \mathtt{1}}
\quad
\dfrac{\mathtt{0}\text{ е стойност}}{\mathtt{0} \Downarrow^{0}_{\mathtt{nat}} \mathtt{0}}
}{\underbrace{\mathtt{if}\ \mathtt{2\text{-}1\text{-}1\ ==\ 0}\ \mathtt{then}\ \mathtt{0}\ \mathtt{else}\ \ \mathtt{3\ +\ (fix}(\tau_0)\ \mathtt{3\ (2\text{-}1\text{-}1\text{-}1))}}_{\tau_1[\mathtt{y}/\mathtt{2\text{-}1\text{-}1}]} \Downarrow^{4}_{\mathtt{nat}} \mathtt{0}}
$$

Фигура 4.7: Последна част от изчислението, което показва, че $\tau_1[\mathtt{y}/\mathtt{2\text{-}1\text{-}1}] \Downarrow^{4}_{\mathtt{nat}} \mathtt{0}$.

4.4 Денотационна семантика

Семантиката на всеки тип ще бъде област на Скот както следва:

$$[\![\texttt{nat}]\!] \stackrel{\text{деф}}{=} \mathbb{N}_\perp$$
$$[\![\texttt{a} \to \texttt{b}]\!] \stackrel{\text{деф}}{=} [[\![\texttt{a}]\!] \stackrel{\text{н}}{\to} [\![\texttt{b}]\!]].$$

За един типов контекст Γ, дефинираме $[\![\Gamma]\!]$ по следния начин:

Да напомним, че

$$\emptyset_\perp = (\{\perp\}, \sqsubseteq, \perp).$$

- Ако $\Gamma = \emptyset$, то $[\![\Gamma]\!] = \emptyset_\perp$;
- Ако $\Gamma = \Gamma', \texttt{x}\!:\!\texttt{a}$, то $[\![\Gamma]\!] = [\![\Gamma']\!] \times [\![\texttt{a}]\!]$.

От Раздел 1.2.2 знаем, че $[\![\Gamma]\!]$ е област на Скот.

Например, ако $\Gamma = \texttt{x}_1 : \texttt{a}_1,\ \texttt{x}_2 : \texttt{a}_2,\ \texttt{x}_3 : \texttt{a}_3$, то

$$[\![\Gamma]\!] = ([\![\texttt{a}_1]\!] \times [\![\texttt{a}_2]\!]) \times [\![\texttt{a}_3]\!].$$

Сега трябва да дефинираме семантика на термовете. За всеки терм, за който $\texttt{fv}(\tau) \subseteq \texttt{dom}(\Gamma)$ и за произволни $\overline{u} \in [\![\Gamma]\!]$, дефинираме неговата стойност $[\![\tau]\!]_\Gamma(\overline{u})$ по следния начин:

- Нека $\tau \equiv \texttt{n}$. Тогава
$$[\![\texttt{n}]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} n.$$
- Нека $\tau \equiv \texttt{x}_i$. Тогава
$$[\![\texttt{x}_i]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} u_i.$$

За `(+)` вижте Раздел 3.2.1.

- Нека $\tau \equiv \tau_1 + \tau_2$. Тогава
$$[\![\tau_1 + \tau_2]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} \texttt{(+)}([\![\tau_1]\!]_\Gamma(\overline{u}), [\![\tau_2]\!]_\Gamma(\overline{u})).$$
- Нека $\tau \equiv \tau_1 - \tau_2$. Тогава
$$[\![\tau_1 - \tau_2]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} \texttt{(-)}([\![\tau_1]\!]_\Gamma(\overline{u}), [\![\tau_2]\!]_\Gamma(\overline{u})).$$

За `(==)` вижте Раздел 3.2.1.

- Нека $\tau \equiv \tau_1 \texttt{ == } \tau_2$. Тогава
$$[\![\tau_1 \texttt{ == } \tau_2]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} \texttt{(==)}([\![\tau_1]\!]_\Gamma(\overline{u}), [\![\tau_2]\!]_\Gamma(\overline{u})).$$

За `if` вижте *Задача* 1.9.

- Нека $\tau \equiv \texttt{if } \tau_1 \texttt{ then } \tau_2 \texttt{ else } \tau_3$. Тогава
$$[\![\texttt{if } \tau_1 \texttt{ then } \tau_2 \texttt{ else } \tau_3]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} \texttt{if}([\![\tau_1]\!]_\Gamma(\overline{u}), [\![\tau_2]\!]_\Gamma(\overline{u}), [\![\tau_3]\!]_\Gamma(\overline{u})).$$

За `eval` вижте *Задача* 1.13.

- Нека $\tau \equiv \tau_1\tau_2$. Тогава
$$[\![\tau_1\tau_2]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} \texttt{eval}([\![\tau_1]\!]_\Gamma(\overline{u}), [\![\tau_2]\!]_\Gamma(\overline{u})).$$

За $\mathtt{lfp}$ вижте Раздел 1.6.

- Нека $\tau \equiv \mathtt{fix}(\tau')$. Тогава

$$[\![\mathtt{fix}(\tau')]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} \mathtt{lfp}([\![\tau']\!]_\Gamma(\overline{u})).$$

За $\mathtt{curry}$ вижте *Дефиниция* 1.4.

- Нека $\tau \equiv \lambda\, \mathtt{y} : \mathtt{b}\,.\, \tau'$, като $\mathtt{y} \notin \mathtt{dom}(\Gamma)$. Нека $\Gamma' \stackrel{\text{деф}}{=} \Gamma, \mathtt{y} : \mathtt{b}$. Тогава

$$[\![\lambda\, \mathtt{y} : \mathtt{b}\,.\, \tau']\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} \mathtt{curry}([\![\tau']\!]_{\Gamma'})(\overline{u}).$$

Не е ясно дали винаги горните дефиниции имат смисъл. Например, не е ясно какво изображение съответства на затворения терм $1 + (\lambda\, \mathtt{x} : \mathtt{nat}\,.\, x)$. Сега ще докажем, че винаги, когато един терм е добре типизиран, то горната дефиниция има смисъл.

Лема 4.3. Ако $\Gamma \vdash \tau : \mathtt{a}$, то $[\![\tau]\!]_\Gamma \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [\![\mathtt{a}]\!]]$.

Доказателство. Доказателството протича с индукция по построението на термовете като съществено използваме *Твърдение* 1.8 според което, ако $f \in [\mathcal{A} \xrightarrow{\text{н}} \mathcal{B}]$ и $g \in [\mathcal{B} \xrightarrow{\text{н}} \mathcal{C}]$, то $g \circ f \in [\mathcal{A} \xrightarrow{\text{н}} \mathcal{C}]$.

Изображението $f \times g$ е дефинирано в *Твърдение* 1.9.

- Нека $\tau \equiv \mathtt{n}$. Щом $\Gamma \vdash \tau : \mathtt{a}$, то по правилата за типизиране следва, че $\mathtt{a} = \mathtt{nat}$. Сега лесно се съобразява, че изображението $[\![\mathtt{n}]\!]_\Gamma \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [\![\mathtt{nat}]\!]]$, където $[\![\mathtt{n}]\!]_\Gamma(\overline{u}) = n$. Това е така, защото за всяка верига $(\overline{u}_i)_{i=0}^\infty$ от елементи на $[\![\Gamma]\!]$,

$$[\![\mathtt{n}]\!]_\Gamma(\bigsqcup_i \overline{u}_i) = n = \bigsqcup_i \{[\![\mathtt{n}]\!](\overline{u}_i)\}.$$

- Нека $\tau \equiv \mathtt{x}_i$. Щом $\Gamma \vdash \tau : \mathtt{a}$, то по правилата за типизиране следва, че $\mathtt{a} = \mathtt{a}_i$. Сега лесно се съобразява, че изображението $[\![\mathtt{x}_i]\!]_\Gamma \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [\![\mathtt{a}_i]\!]]$, където $[\![\mathtt{x}_i]\!]_\Gamma(\overline{u}) = u_i$. Това е така, защото за всяка верига $(\overline{u}_n)_{n=0}^\infty$ от елементи на $[\![\Gamma]\!]$,

$$[\![\mathtt{x}_i]\!]_\Gamma(\bigsqcup_n \overline{u}_n) = \bigsqcup_n u_{i,n} = \bigsqcup_i \{[\![\mathtt{x}_i]\!](\overline{u}_n)\}.$$

Тук означаваме

$$\overline{u}_k = (u_{1,k}, \ldots, u_{n,k}).$$

- Нека $\tau \equiv \tau_1 + \tau_2$. Щом $\Gamma \vdash \tau : \mathtt{a}$, то по правилата за типизиране следва, че $\mathtt{a} = \mathtt{nat}$, а също и $\Gamma \vdash \tau_1 : \mathtt{nat}$ и $\Gamma \vdash \tau_2 : \mathtt{nat}$. От **(И.П.)** имаме, че

$$[\![\tau_1]\!]_\Gamma \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [\![\mathtt{nat}]\!]];$$
$$[\![\tau_2]\!]_\Gamma \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [\![\mathtt{nat}]\!]].$$

Това означава, че $([\![\tau_1]\!] \times [\![\tau_2]\!]) \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [\![\mathtt{nat}]\!] \times [\![\mathtt{nat}]\!]]$. Тогава имаме следното равенство

$$[\![\tau_1 + \tau_2]\!]_\Gamma = (\mathtt{+}) \circ ([\![\tau_1]\!] \times [\![\tau_2]\!]) \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [\![\mathtt{nat}]\!]],$$

Използваме, че композиция на непрекъснати изображения е непрекъснато изображение. За произволни $\overline{u} \in [\![\Gamma]\!]$,

$$((\mathtt{+}) \circ ([\![\tau_1]\!] \times [\![\tau_2]\!]))(\overline{u}) = (\mathtt{+})(([\![\tau_1]\!] \times [\![\tau_2]\!])(\overline{u})) = (\mathtt{+})([\![\tau_1]\!]_\Gamma(\overline{u}), [\![\tau_2]\!]_\Gamma(\overline{u})) \stackrel{\text{деф}}{=} [\![\tau]\!]_\Gamma(\overline{u}).$$

- Нека $\tau \equiv \tau_1$ `==` τ_2. Съобразете сами, че

$$[\![\tau_1 \texttt{==} \tau_2]\!]_\Gamma = (\texttt{==}) \circ ([\![\tau_1]\!]_\Gamma \times [\![\tau_2]\!]_\Gamma) \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [\![\mathtt{a}]\!]].$$

- Нека $\tau \equiv \mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3$. Съобразете сами, че

$$[\![\mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3]\!]_\Gamma = \mathtt{if} \circ ([\![\tau_1]\!]_\Gamma \times [\![\tau_2]\!]_\Gamma \times [\![\tau_3]\!]_\Gamma) \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [\![\mathtt{a}]\!]].$$

- Нека $\tau \equiv \tau_1 \tau_2$. Щом $\Gamma \vdash \tau_1 \tau_2 : \mathtt{a}$, то от правилата за типизиране следва, че

$$\Gamma \vdash \tau_1 : \mathtt{b} \to \mathtt{a}$$
$$\Gamma \vdash \tau_2 : \mathtt{b}.$$

  От **(И.П.)** за τ_1 и τ_2 знаем, че

$$[\![\tau_1]\!]_\Gamma \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [[\![\mathtt{b}]\!] \xrightarrow{\text{н}} [\![\mathtt{a}]\!]]]$$
$$[\![\tau_2]\!]_\Gamma \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [\![\mathtt{b}]\!]]$$

  Оттук получаваме, че за произволни $\overline{u} \in [\![\Gamma]\!]$,

$$[\![\tau_1]\!]_\Gamma(\overline{u}) \in [[\![\mathtt{b}]\!] \xrightarrow{\text{н}} [\![\mathtt{a}]\!]]$$
$$[\![\tau_2]\!]_\Gamma(\overline{u}) \in [\![\mathtt{b}]\!].$$

  Тогава

$$[\![\tau_1 \tau_2]\!]_\Gamma = \mathtt{eval} \circ ([\![\tau_1]\!]_\Gamma \times [\![\tau_2]\!]_\Gamma) \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [\![\mathtt{a}]\!]],$$

  За произволни $\overline{u} \in [\![\Gamma]\!]$,

  $(\mathtt{eval} \circ ([\![\tau_1]\!]_\Gamma \times [\![\tau_2]\!]_\Gamma))(\overline{u}) =$
  $\mathtt{eval}(([\![\tau_1]\!]_\Gamma \times [\![\tau_2]\!]_\Gamma)(\overline{u})) =$
  $\mathtt{eval}([\![\tau_1]\!]_\Gamma(\overline{u}), [\![\tau_2]\!]_\Gamma(\overline{u})) \stackrel{\text{деф}}{=}$
  $[\![\tau_1 \tau_2]\!]_\Gamma(\overline{u})$.

- Нека сега $\tau \equiv \mathtt{fix}(\tau')$. Понеже $\Gamma \vdash \mathtt{fix}(\tau') : \mathtt{a}$, то от правилата за типизиране имаме, че $\Gamma \vdash \tau' : \mathtt{a} \to \mathtt{a}$. От **(И.П.)** знаем, че

$$[\![\tau']\!]_\Gamma \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [[\![\mathtt{a}]\!] \xrightarrow{\text{н}} [\![\mathtt{a}]\!]]].$$

  Това означава, че за произволни $\overline{u} \in [\![\Gamma]\!]$,

$$[\![\tau']\!]_\Gamma(\overline{u}) \in [[\![\mathtt{a}]\!] \xrightarrow{\text{н}} [\![\mathtt{a}]\!]].$$

  Следователно $[\![\tau']\!]_\Gamma(\overline{u})$ е изображение, което според *Теорема* 1.5 притежава най-малка неподвижна точка. Тогава

$$[\![\mathtt{fix}(\tau')]\!]_\Gamma = Y \circ [\![\tau']\!]_\Gamma \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [\![\mathtt{a}]\!]],$$

  Непрекъснатото изображението Y е дефинирано *Теорема* 1.6. За произволни $\overline{u} \in [\![\Gamma]\!]$,

  $(Y \circ [\![\tau']\!]_\Gamma)(\overline{u}) =$
  $Y([\![\tau']\!]_\Gamma(\overline{u})) =$
  $\mathtt{lfp}([\![\tau']\!]_\Gamma(\overline{u})) \stackrel{\text{деф}}{=}$
  $[\![\mathtt{fix}(\tau')]\!]_\Gamma(\overline{u})$.

- Нека $\tau \equiv \lambda\, \mathtt{y} : \mathtt{b}\, .\, \tau'$, като $\mathtt{y} \not\in \mathtt{dom}(\Gamma)$. Щом $\Gamma \vdash \lambda\, \mathtt{y} : \mathtt{b}\, .\, \tau' : \mathtt{a}$, то от правилата за типизиране следва, че $\mathtt{a} = \mathtt{b} \to \mathtt{c}$ и

$$\Gamma, \mathtt{y} : \mathtt{b} \vdash \tau' : \mathtt{c}.$$

  Нека $\Gamma' = \Gamma, \mathtt{y} : \mathtt{b}$. Тогава $[\![\Gamma']\!] = [\![\Gamma]\!] \times [\![\mathtt{b}]\!]$, а от **(И.П.)** имаме, че

$$[\![\tau']\!]_{\Gamma'} \in [[\![\Gamma]\!] \times [\![\mathtt{b}]\!] \xrightarrow{\text{н}} [\![\mathtt{c}]\!]].$$

  Тогава от *Твърдение* 1.10 следва, че

$$[\![\lambda\, \mathtt{y} : \mathtt{b}\, .\, \tau]\!]_\Gamma \stackrel{\text{деф}}{=} \mathtt{curry}([\![\tau']\!]_{\Gamma'}) \in [[\![\Gamma]\!] \xrightarrow{\text{н}} [[\![\mathtt{b}]\!] \xrightarrow{\text{н}} [\![\mathtt{c}]\!]]].$$

$\square$

Забележка 4.1. За $\Gamma = \emptyset$, ще пишем $[\![\tau]\!]$ вместо $[\![\tau]\!]_\emptyset(\bot)$. Формално погледнато, $[\![\tau]\!]_\emptyset \in [\emptyset_\bot \overset{\text{н}}{\to} \mathcal{A}]$, за някоя област на Скот $\mathcal{A}$. Но ние знаем, че $[\emptyset_\bot \overset{\text{н}}{\to} \mathcal{A}] \cong \mathcal{A}$. Следователно, можем да считаме, че $[\![\tau]\!] \in \mathcal{A}$. В противен случай, трябва винаги да пишем $[\![\tau]\!]_\emptyset(\bot)$ вместо $[\![\tau]\!]$, което би било досадно.

Твърдение 4.5. Нека имаме следните типови контексти:

$$\Gamma = \mathtt{x}_1 : \mathtt{a}_1, \ldots, \mathtt{x}_i : \mathtt{a}_i, \ldots, \mathtt{x}_j : \mathtt{a}_j, \ldots, \mathtt{x}_n : \mathtt{a}_n;$$
$$\Delta = \mathtt{x}_1 : \mathtt{a}_1, \ldots, \mathtt{x}_j : \mathtt{a}_j, \ldots, \mathtt{x}_i : \mathtt{a}_i, \ldots, \mathtt{x}_n : \mathtt{a}_n,$$

т.е. Δ се получава от Γ като разменим местата на i-тата и j-тата двойка. Тогава за всеки терм τ, такъв че $\Gamma \vdash \tau : \mathtt{a}$, е изпълнено, че $\Delta \vdash \tau : \mathtt{a}$ и за всеки $(u_1, \ldots, u_n) \in [\![\Gamma]\!]$,

$$[\![\tau]\!]_\Gamma(u_1, \ldots, u_i, \ldots, u_j, \ldots, u_n) = [\![\tau]\!]_\Delta(u_1, \ldots, u_j, \ldots, u_i, \ldots, u_n).$$

Ясно е, че това твърдение се обобщава за произволна пермутация на индекстите $1, \ldots, n$ [10, стр. 106].

Упътване. Индукция по построението на терма τ. $\square$

Понеже дефинирахме релацията $\leq_{op}$, то естествено е да дефинираме и аналогична релация $\leq_{den}$.

Определение 4.4. За всеки два затворени терма τ_1 и τ_2 от тип $\mathtt{a}$, дефинираме релацията $\tau_1 \leq_{den} \tau_2 : \mathtt{a}$ по следния начин:

- $\emptyset \vdash \tau_1 : \mathtt{a}$ и $\emptyset \vdash \tau_2 : \mathtt{a}$,
- $[\![\tau_1]\!] \sqsubseteq [\![\tau_2]\!]$.

Лесно се съобразява, че релацията $\leq_{den}$ е рефлексивна и транзитивна. Тя поражда релация на еквивалентност $\cong_{den}$ като дефинираме $\tau_1 \cong_{den} \tau_2 : \mathtt{a}$ точно тогава, когато $\tau_1 \leq_{den} \tau_2 : \mathtt{a}$ и $\tau_2 \leq_{den} \tau_1 : \mathtt{a}$.

Пример 4.2. Нека $\mathtt{a} = \mathtt{nat} \to (\mathtt{nat} \to \mathtt{nat})$ и

$$\tau \equiv \mathtt{fix(}\underbrace{\lambda\,\mathtt{f}:\mathtt{a}\,.\,\overbrace{\lambda\,\mathtt{x}:\mathtt{nat}\,.\,\lambda\,\mathtt{y}:\mathtt{nat}\,.\,\mathtt{if}\ \mathtt{y\ ==\ 0}\ \mathtt{then}\ 0\ \mathtt{else}\ \mathtt{x\ +\ (f\ x\ (y\text{-}1))}}^{\tau_0'}}_{\tau_0}\mathtt{)}.$$

Знаем, че $[\![\tau]\!] \in [\mathbb{N}_\perp \overset{\text{н}}{\to} [\mathbb{N}_\perp \overset{\text{н}}{\to} \mathbb{N}_\perp]]$, където

$$[\![\tau]\!] \overset{\text{деф}}{=} \mathtt{lfp}([\![\tau_0]\!]).$$

Ясно е, че $[\![\tau_0]\!] \in [[\![\mathtt{a}]\!] \overset{\text{н}}{\to} [\![\mathtt{a}]\!]]$ и $[\![\tau_0]\!] = \mathtt{curry}([\![\tau_0']\!]_{\mathtt{f:a}}) = [\![\tau_0']\!]_{\mathtt{f:a}}$ и сега пък ако положим

$$\tau_0'' \equiv \lambda\,\mathtt{y}:\mathtt{nat}\,.\,\mathtt{if}\ \mathtt{y\ ==\ 0}\ \mathtt{then}\ 0\ \mathtt{else}\ \mathtt{x\ +\ (f\ x\ (y\text{-}1))},$$
$$\tau_0''' \equiv \mathtt{if}\ \mathtt{y\ ==\ 0}\ \mathtt{then}\ 0\ \mathtt{else}\ \mathtt{x\ +\ (f\ x\ (y\text{-}1))}.$$

то ще получим, че

$$[\![\tau_0']\!]_{\mathtt{f:a}} = \mathtt{curry}([\![\tau_0'']\!]_{\mathtt{f:a,x:nat}}),$$

и тогава

$$[\![\tau_0']\!]_{\mathtt{f:a}}(\varphi)(m) = [\![\tau_0'']\!]_{\mathtt{f:a,x:nat}}(\varphi, m).$$

Сега вече получаваме, че

$$[\![\tau_0'']\!]_{\mathtt{f:a,x:nat}} = \mathtt{curry}([\![\tau_0''']\!]_{\mathtt{f:a,x:nat,y:nat}}),$$

т.е.

$$[\![\tau_0'']\!]_{\mathtt{f:a,x:nat}}(\varphi, m)(n) = [\![\tau_0''']\!]_{\mathtt{f:a,x:nat,y:nat}}(\varphi, m, n).$$

Обединявайки всичко получаваме, че:

$$\begin{aligned}
[\![\tau_0]\!](\varphi)(m)(n) &= [\![\tau_0']\!]_{\mathtt{f:a}}(\varphi)(m)(n)\\
&= [\![\tau_0'']\!]_{\mathtt{f:a,x:nat}}(\varphi, m)(n)\\
&= [\![\tau_0''']\!]_{\mathtt{f:a,x:nat,y:nat}}(\varphi, m, n)\\
&= [\![\mathtt{if\ y{=}{=}0\ then}\ 0\ \mathtt{else\ x\ +\ (f\ x\ (y\text{-}1))}]\!](\varphi, m, n)\\
&= \mathtt{if}([\![\mathtt{y{=}{=}0}]\!](\varphi, m, n), [\![0]\!](\varphi, m, n), [\![\mathtt{x\ +\ (f\ x\ (y\text{-}1))}]\!](\varphi, m, n)).
\end{aligned}$$

Накрая получаваме, че

$$[\![\tau_0]\!](\varphi)(m)(n) = \begin{cases} 0, & \text{ако } n = 0\\ \mathtt{(+)}(m, \varphi(m)(n-1)), & \text{ако } n > 0\\ \perp, & \text{ако } n = \perp. \end{cases}$$

Сега вече знаем как по теоремата на Клини да докажем, че

$$\mathtt{lfp}([\![\tau_0]\!])(m)(n) = \begin{cases} m * n, & \text{ако } m, n \in \mathbb{N}\\ \perp, & \text{иначе} \end{cases}$$

От *Лема* 4.1 вече знаем, че типизиращата релация е съвместима с операцията субституция. Сега ще видим, че денотационната семантика е съвместима с операцията субституция.

Лема 4.4 (Лема за замяната). Нека $\Gamma \vdash \rho : \mathtt{a}$ и $\Gamma, \mathtt{x} : \mathtt{a} \vdash \tau : \mathtt{b}$, като $\mathtt{x} \in \mathtt{dom}(\Gamma)$. Тогава за всяко $\overline{u} \in [\![\Gamma]\!]$ имаме равенството

$$[\![\tau[\mathtt{x}/\rho]]\!]_{\Gamma}(\overline{u}) = [\![\tau]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_{\Gamma}(\overline{u})).$$

Защо да не взема ρ да бъде затворен терм ?

Доказателство. Индукция по построението на термовете. Започваме с базовият случай, който може да се разбие на три подслучая.

- Нека $\tau \equiv \mathtt{n}$. Тогава

$$\begin{aligned}
[\![\tau[\mathtt{x}/\rho]]\!]_{\Gamma}(\overline{u}) &= [\![\mathtt{n}[\mathtt{x}/\rho]]\!]_{\Gamma}(\overline{u}) \\
&= [\![\mathtt{n}]\!]_{\Gamma}(\overline{u}) \\
&= n \\
&= [\![\mathtt{n}]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, v) && \text{// за каквото и да е } v \\
&= [\![\mathtt{n}]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_{\Gamma}(\overline{u})) \\
&= [\![\tau]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_{\Gamma}(\overline{u})).
\end{aligned}$$

- Нека $\tau \equiv \mathtt{x}_i$, където $\mathtt{x}_i \not\equiv \mathtt{x}$. Тогава

$$\begin{aligned}
[\![\tau[\mathtt{x}/\rho]]\!]_{\Gamma}(\overline{u}) &= [\![\mathtt{x}_i[\mathtt{x}/\rho]]\!]_{\Gamma}(\overline{u}) \\
&= [\![\mathtt{x}_i]\!]_{\Gamma}(\overline{u}) \\
&= u_i \\
&= [\![\mathtt{x}_i]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, v) && \text{// за каквото и да е } v \\
&= [\![\mathtt{x}_i]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_{\Gamma}(\overline{u})) \\
&= [\![\tau]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_{\Gamma}(\overline{u})).
\end{aligned}$$

- Нека $\tau \equiv \mathtt{x}$. Тогава

$$\begin{aligned}
[\![\tau[\mathtt{x}/\rho]]\!]_{\Gamma}(\overline{u}) &= [\![\mathtt{x}[\mathtt{x}/\rho]]\!]_{\Gamma}(\overline{u}) \\
&= [\![\rho]\!]_{\Gamma}(\overline{u}) \\
&= [\![\mathtt{x}]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_{\Gamma}(\overline{u})) \\
&= [\![\tau]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_{\Gamma}(\overline{u})).
\end{aligned}$$

Сега преминаваме към индукционната стъпка.

- Нека $\tau \equiv \tau_1 + \tau_2$. Тогава

$$[\![\tau]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_{\Gamma}(\overline{u})) = [\![\tau_1 + \tau_2]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_{\Gamma}(\overline{u}))$$

$$\begin{aligned}
&= (+)([\![\tau_1]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u})), [\![\tau_1]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u})))\\
&= (+)([\![\tau_1[\mathtt{x}/\rho]]\!]_\Gamma(\overline{u}), [\![\tau_2[\mathtt{x}/\rho]]\!]_\Gamma(\overline{u})) && \text{// \textbf{(И.П.)} за } \tau_1 \text{ и } \tau_2\\
&= [\![\tau_1[\mathtt{x}/\rho] + \tau_2[\mathtt{x}/\rho]]\!]_\Gamma(\overline{u})\\
&= [\![\tau[\mathtt{x}/\rho]]\!]_\Gamma(\overline{u}).
\end{aligned}$$

- Нека $\tau \equiv \tau_1 - \tau_2$.
- Нека $\tau \equiv \tau_1 == \tau_2$.
- Нека $\tau \equiv \mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3$.
- Нека $\tau \equiv \tau_1\tau_2$. Тогава

$$\begin{aligned}
[\![\tau]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u})) &= [\![\tau_1\tau_2]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u}))\\
&\stackrel{\text{деф}}{=} \mathtt{eval}([\![\tau_1]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u})), [\![\tau_2]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u}))) && \text{// \textit{Дефиниция} 1.3}\\
&= \mathtt{eval}([\![\tau_1[\mathtt{x}/\rho]]\!]_\Gamma(\overline{u}), [\![\tau_2[\mathtt{x}/\rho]]\!]_\Gamma(\overline{u})) && \text{// \textbf{(И.П.)} за } \tau_1 \text{ и } \tau_2\\
&= [\![\tau_1[\mathtt{x}/\rho] (\tau_2[\mathtt{x}/\rho])]\!]_\Gamma(\overline{u})\\
&= [\![\tau[\mathtt{x}/\rho]]\!]_\Gamma(\overline{u}).
\end{aligned}$$

- Нека $\tau \equiv \mathtt{fix}(\tau')$.

  Тогава получаваме следната верига от равенства:

$$\begin{aligned}
[\![\tau[\mathtt{x}/\rho]]\!]_\Gamma(\overline{u}) &= [\![\mathtt{fix}(\tau'[\mathtt{x}/\rho])]\!]_\Gamma(\overline{u})\\
&= \mathtt{lfp}([\![\tau'[\mathtt{x}/\rho]]\!]_\Gamma(\overline{u})) && \text{// от деф.}\\
&= \mathtt{lfp}([\![\tau']\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u}))) && \text{// от \textbf{(И.П.)} за } \tau'\\
&= [\![\mathtt{fix}(\tau')]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u}))\\
&= [\![\tau]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u})).
\end{aligned}$$

Тук е важно, че

$$[\![\Delta]\!] = [\![\Gamma]\!] \times [\![\mathtt{a}_n]\!].$$

- Нека $\tau \equiv \lambda\,\mathtt{y} : \mathtt{c}\,.\,\tau'$, където $\mathtt{y} \notin \mathtt{dom}(\Gamma) \cup \{\mathtt{x}\}$.

  Тогава, понеже имаме, че $\Gamma, \mathtt{y} : \mathtt{c} \vdash \tau'[\mathtt{x}/\rho] : \mathtt{d}$, то можем да приложим **(И.П.)** за τ' и така получаваме, че за всяко $\overline{u} \in [\![\Gamma]\!]$ и $v \in [\![\mathtt{c}]\!]$,

$$\begin{aligned}
[\![\tau[\mathtt{x}/\rho]]\!]_\Gamma(\overline{u})(v) &= [\![\lambda\,\mathtt{y} : \mathtt{c}\,.\,\tau'[\mathtt{x}/\rho]]\!]_\Gamma(\overline{u})(v)\\
&= \mathtt{curry}([\![\tau'[\mathtt{x}/\rho]]\!]_{\Gamma,\mathtt{y:c}})(\overline{u})(v)\\
&\stackrel{\text{деф}}{=} [\![\tau'[\mathtt{x}/\rho]]\!]_{\Gamma,\mathtt{y:c}}(\overline{u}, v) && \text{// \textit{Дефиниция} 1.4}\\
&= [\![\tau']\!]_{\Gamma,\mathtt{y:c},\mathtt{x:a}}(\overline{u}, v, [\![\rho]\!]_{\Gamma,\mathtt{y:c}}(\overline{u}, v)) && \text{// \textbf{(И.П.)}}\\
&= [\![\tau']\!]_{\Gamma,\mathtt{y:c},\mathtt{x:a}}(\overline{u}, v, [\![\rho]\!]_\Gamma(\overline{u})) && \text{// } \mathtt{fv}(\rho) \subseteq \mathtt{dom}(\Gamma)\\
&= [\![\tau']\!]_{\Gamma,\mathtt{x:a},\mathtt{y:c}}(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u}), v)\\
&= \mathtt{curry}([\![\tau']\!]_{\Gamma,\mathtt{x:a},\mathtt{y:c}})(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u}))(v)\\
&= [\![\lambda\,\mathtt{y} : \mathtt{c}\,.\,\tau']\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u}))(v)\\
&= [\![\tau]\!]_{\Gamma,\mathtt{x:a}}(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u}))(v).
\end{aligned}$$

Така получихме, че

$$[\![\tau[\mathtt{x}/\rho]]\!]_\Gamma(\overline{u}) = [\![\tau]\!]_{\Gamma,\mathtt{x}:\mathtt{a}}(\overline{u}, [\![\rho]\!]_\Gamma(\overline{u})).$$

$\square$

4.5 Един важен пример

Добре е още тук да разгледаме термове, които ще ни бъдат полезни понататък, когато искаме да изучим свойствата на операционната и денотационната семантики на PCF.

Определение 4.5. За произволен тип $\mathtt{a}$, да означим затворените термове

$$\Omega_{\mathtt{a}} \stackrel{\text{деф}}{=} \mathtt{fix}(\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mathtt{x})$$
$$\Omega'_{\mathtt{a}} \stackrel{\text{деф}}{=} \lambda\, \mathtt{x} : \mathtt{a} \,.\, \mathtt{fix}(\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mathtt{x}).$$

✍ Студентите би трябвало да могат да докажат твърденията в *Задача* 4.2 сами!

Задача 4.2. Докажете, че за всеки тип $\mathtt{a}$ са изпълнени следните свойства:

(1) $\emptyset \vdash \Omega_{\mathtt{a}} : \mathtt{a}$ и $\emptyset \vdash \Omega'_{\mathtt{a}} : \mathtt{a} \to \mathtt{a}$;

(2) $\Omega_{\mathtt{a}} \not\Downarrow_{\mathtt{a}}$ и $\Omega'_{\mathtt{a}} \Downarrow^0_{\mathtt{a}\to\mathtt{a}} \Omega'_{\mathtt{a}}$;

(3) $[\![\Omega_{\mathtt{a}}]\!] = \bot^{[\![\mathtt{a}]\!]}$ и $[\![\Omega'_{\mathtt{a}}]\!] = [\![\Omega_{\mathtt{a}\to\mathtt{a}}]\!] = \bot^{[\![\mathtt{a}\to\mathtt{a}]\!]}$.

Упътване. Доказателството на Свойство (1) представлява едно просто упражнение, което за пълнота на изложението ще направим.

$$\dfrac{\dfrac{\mathtt{x} : \mathtt{a} \vdash \mathtt{x} : \mathtt{a}}{\emptyset \vdash \lambda\, \mathtt{x} : \mathtt{a} \,.\, \mathtt{x} : \mathtt{a} \to \mathtt{a}}}{\emptyset \vdash \underbrace{\mathtt{fix}(\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mathtt{x})}_{\Omega_{\mathtt{a}}} : \mathtt{a}} \qquad \dfrac{\dfrac{\dfrac{\text{вече доказано}}{\emptyset \vdash \Omega_{\mathtt{a}} : \mathtt{a}}}{\mathtt{x} : \mathtt{a} \vdash \Omega_{\mathtt{a}} : \mathtt{a}}}{\underbrace{\emptyset \vdash \lambda\, \mathtt{x} : \mathtt{a} \,.\, \Omega_{\mathtt{a}}}_{\Omega'_{\mathtt{a}}} : \mathtt{a} \to \mathtt{a}}$$

Втората част на Свойство (2) не заслужава внимание, защото $\Omega'_{\mathtt{a}}$ е стойност и следователно по дефиниция $\Omega'_{\mathtt{a}} \Downarrow^0_{\mathtt{a}\to\mathtt{a}} \Omega'_{\mathtt{a}}$.

За първата част на Свойство (2), да допуснем, че $\Omega_{\mathtt{a}} \Downarrow_{\mathtt{a}} \mathtt{v}$, за някоя стойност $\mathtt{v} : \mathtt{a}$, и нека фиксираме ℓ да бъде *най-малкият* брой стъпки, за които $\Omega_{\mathtt{a}} \Downarrow^{\ell}_{\mathtt{a}} \mathtt{v}$. Но тогава

$$\dfrac{\dfrac{\dfrac{\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mathtt{x} \text{ е стойност}}{\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mathtt{x} \Downarrow^0_{\mathtt{a}\to\mathtt{a}} \lambda\, \mathtt{x} : \mathtt{a} \,.\, \mathtt{x}} \qquad \dfrac{\Omega_{\mathtt{a}} \Downarrow^{\ell-2}_{\mathtt{a}} \mathtt{v}}{\mathtt{x}[\mathtt{x}/\mathtt{fix}(\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mathtt{x})] \Downarrow^{\ell-2}_{\mathtt{a}} \mathtt{v}}}{(\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mathtt{x})\mathtt{fix}(\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mathtt{x}) \Downarrow^{\ell-1}_{\mathtt{a}} \mathtt{v}}\,(\text{cbn})}{\underbrace{\mathtt{fix}(\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mathtt{x})}_{\Omega_{\mathtt{a}}} \Downarrow^{\ell}_{\mathtt{a}} \mathtt{v}}\,(\text{fix})$$

Получихме, че $\Omega_{\mathtt{a}} \Downarrow^{\ell-2}_{\mathtt{a}} \mathtt{v}$, което е противоречие с минималността на ℓ.

Сега да разгледаме първата част на Свойство (3). За произволен тип $\mathtt{a}$, да означим с $\mathtt{id}_{[\![\mathtt{a}]\!]}$ функцията идентитет за областта на Скот $[\![\mathtt{a}]\!]$, т.е. $\mathtt{id}_{[\![\mathtt{a}]\!]}(x) = x$ за всяко $x \in [\![\mathtt{a}]\!]$. Имаме, че:

Да напомним, че $f^0 \stackrel{\text{деф}}{=} id$ и $f^{n+1} \stackrel{\text{деф}}{=} f \circ f^n$. В нашия случай, $f = id$ и следователно $id^n = id$ за всяко n.

$$\begin{aligned}
[\![\Omega_{\mathtt{a}}]\!] &= [\![\mathtt{fix}(\lambda\,\mathtt{x} : \mathtt{a}\,.\,\mathtt{x})]\!] \\
&= \mathtt{lfp}([\![\lambda\,\mathtt{x} : \mathtt{a}\,.\,\mathtt{x}]\!]) \\
&= \mathtt{lfp}(\mathtt{id}_{[\![\mathtt{a}]\!]}) && /\!/ \ [\![\lambda\,\mathtt{x} : \mathtt{a}\,.\,\mathtt{x}]\!] = \mathtt{id}_{[\![\mathtt{a}]\!]} \\
&= \bigsqcup_n \mathtt{id}^n_{[\![\mathtt{a}]\!]}(\bot^{[\![\mathtt{a}]\!]}) && /\!/ \ \text{Теоремата на Клини} \\
&= \bigsqcup_n \bot^{[\![\mathtt{a}]\!]} && /\!/ \ \mathtt{id}^n(\bot^{[\![\mathtt{a}]\!]}) = \bot^{[\![\mathtt{a}]\!]} \\
&= \bot^{[\![\mathtt{a}]\!]}.
\end{aligned}$$

Сега преминаваме към втората част на Свойство (3). За произволен елемент $u \in [\![\mathtt{a}]\!]$,

Да напомним, че за всяко u, $\bot^{[\![\mathtt{a}\to\mathtt{a}]\!]}(u) \stackrel{\text{деф}}{=} \bot^{[\![\mathtt{a}]\!]}$.

$$\begin{aligned}
[\![\Omega'_a]\!](u) &= [\![\lambda\,\mathtt{x} : \mathtt{a}\,.\,\Omega_{\mathtt{a}}]\!](u) \\
&= \mathtt{curry}([\![\Omega_{\mathtt{a}}]\!]_{\mathtt{x:a}})(u) \\
&= [\![\Omega_{\mathtt{a}}]\!]_{\mathtt{x:a}}(u) \\
&= [\![\Omega_{\mathtt{a}}]\!] \\
&= \bot^{[\![\mathtt{a}]\!]}.
\end{aligned}$$

Заключаваме, че $[\![\Omega'_a]\!] = \bot^{[\![\mathtt{a}\to\mathtt{a}]\!]}$. □

```
ghci> :set -XScopedTypeVariables
ghci> let fix f = x where x = f x
ghci> let omega = fix(\(x::a->a) -> x)
ghci> :t omega
omega :: a -> a
ghci> let omega' = \(x::a)->fix(\(x::a) -> x)
ghci> :t omega'
omega' :: a -> a
```

4.6 Коректност

Понеже вече имаме дефинирани операционна и денотационна семантика на термовете, следващата стъпка е да разгледаме каква е връзката между тях. В този раздел ще докажем едната (по-лесната) посока.

Да напомним, че когато термът τ е затворен, то ще пишем $[\![\tau]\!]$ вместо $[\![\tau]\!]_{\emptyset}(\perp)$.

На англ. *soundness*.

Теорема 4.1 (Теорема за коректност)**.** За всеки затворен терм $\tau : \mathtt{b}$ и стойност $\mathtt{v} : \mathtt{b}$, е изпълнена импликацията:

$$\tau \Downarrow_{\mathtt{b}} \mathtt{v} \implies [\![\tau]\!] = [\![\mathtt{v}]\!] \in [\![\mathtt{b}]\!].$$

[4, стр. 206], [7, стр. 133].

Доказателство. Индукция по дължината ℓ на извода $\Downarrow_{\mathtt{b}}^{\ell}$ за всеки тип $\mathtt{b}$. Нека $\ell = 0$. Имаме два случая, защото имаме два вида стойности.

- Нека $\tau \equiv \mathtt{n}$. Ясно е, че $\mathtt{b} = \mathtt{nat}$ и от правилата на операционната семантика имаме, че:

$$\frac{}{\tau \Downarrow_{\mathtt{nat}}^{0} \mathtt{n}}\ (\text{val})$$

  От дефиницията на семантика на терм, директно получаваме, че $[\![\tau]\!] = n = [\![\mathtt{n}]\!]$.

- Нека $\tau \equiv \lambda\ \mathtt{x} : \mathtt{c}\ .\ \tau'$. Тогава $\mathtt{b} = \mathtt{a} \to \mathtt{c}$ и от правилата на операционната семантика имаме, че:

$$\frac{}{\tau \Downarrow_{\mathtt{a}\to\mathtt{c}}^{0} \tau}\ (\text{val})$$

  Ясно е, че $[\![\tau]\!] = [\![\tau]\!] \in [\![b]\!]$.

Така доказахме, че

$$\tau \Downarrow_{\mathtt{b}}^{0} \mathtt{v} \implies [\![\tau]\!] = [\![\mathtt{v}]\!] \in [\![\mathtt{b}]\!].$$

Нека сега $\ell > 0$ и да приемем, че имаме следното индукционно предположение:

$$\tau \Downarrow_{\mathtt{b}}^{<\ell} \mathtt{v} \implies [\![\tau]\!] = [\![\mathtt{v}]\!] \in [\![\mathtt{b}]\!].$$

Ще докажем, че

$$\tau \Downarrow_{\mathtt{b}}^{\ell} \mathtt{v} \implies [\![\tau]\!] = [\![\mathtt{v}]\!] \in [\![\mathtt{b}]\!].$$

- Нека $\tau \equiv \tau_1 + \tau_2$. Тогава от правилата на операционната семантика имаме, че:

$$(\ell = \ell_1 + \ell_2 + 1)\ \frac{\tau_1 \Downarrow_{\mathtt{nat}}^{\ell_1} \mathtt{n}_1 \qquad \tau_2 \Downarrow_{\mathtt{nat}}^{\ell_2} \mathtt{n}_2}{\tau_1 + \tau_2 \Downarrow_{\mathtt{nat}}^{\ell_1+\ell_2+1} \mathtt{n},}\ (\text{plus})$$

където $n = n_1 + n_2$. От **(И.П.)** получаваме, че

$$\begin{aligned}
\llbracket \tau_1 \rrbracket &= \llbracket \mathtt{n}_1 \rrbracket = n_1 \\
\llbracket \tau_2 \rrbracket &= \llbracket \mathtt{n}_2 \rrbracket = n_2.
\end{aligned}$$

Тогава

$$\begin{aligned}
\llbracket \tau_1 + \tau_2 \rrbracket &= \mathtt{(+)}(\llbracket \tau_1 \rrbracket, \llbracket \tau_2 \rrbracket) && \text{// от деф.} \\
&= n_1 + n_2 && \text{// \textbf{(И.П.)}} \\
&= n.
\end{aligned}$$

- Случаите $\tau \equiv \tau_1 - \tau_2$ и $\tau \equiv \tau_1 \mathtt{==} \tau_2$ са аналогични. Оставяме ги на читателя.
- Нека $\tau \equiv \mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3$. Тогава от правилата на операционната семантика имаме, че:

$$(\ell = \ell_1 + \ell_2 + 1)\ \frac{\tau_1 \Downarrow^{\ell_1}_{\mathtt{nat}} \mathtt{n}_1 \qquad \tau_2 \Downarrow^{\ell_2}_{\mathtt{a}} \mathtt{v}_2 \qquad \mathtt{n}_1 \not\equiv \mathtt{0}}{\mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3 \Downarrow^{\ell_1+\ell_2+1}_{\mathtt{a}} \mathtt{v}_2,}\ (\mathrm{if}^+)$$

Тогава от **(И.П.)** получаваме, че:

$$\begin{aligned}
\llbracket \tau_1 \rrbracket &= n_1 \\
\llbracket \tau_2 \rrbracket &= \llbracket \mathtt{v}_2 \rrbracket.
\end{aligned}$$

Тогава

$$\begin{aligned}
\llbracket \mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3 \rrbracket &= \mathtt{if}(\llbracket \tau_1 \rrbracket, \llbracket \tau_2 \rrbracket, \llbracket \tau_3 \rrbracket) \\
&= \mathtt{if}(n_1, \llbracket \tau_2 \rrbracket, \llbracket \tau_3 \rrbracket) && \text{// от \textbf{(И.П.)}} \\
&= \llbracket \tau_2 \rrbracket && \text{// от деф. на } \mathtt{if} \\
&= \llbracket \mathtt{v}_2 \rrbracket. && \text{// от \textbf{(И.П.)}}
\end{aligned}$$

Случаят, когато $\mathtt{n}_1 \equiv \mathtt{0}$ е аналогичен.

- Нека $\tau \equiv \tau_1\tau_2$. Тогава от правилата на операционната семантика имаме, че:

$$(\ell = \ell_1 + \ell_2 + 1)\ \frac{\tau_1 \Downarrow^{\ell_1}_{\mathtt{a}\to\mathtt{b}} \lambda\, \mathtt{x} : \mathtt{a}\, .\, \tau_1' \qquad \tau_1'[x/\tau_2] \Downarrow^{\ell_2}_{\mathtt{b}} \mathtt{v}}{\tau_1\tau_2 \Downarrow^{\ell_1+\ell_2+1}_{\mathtt{b}} \mathtt{v}}\ (\mathrm{cbn})$$

Тогава от **(И.П.)** получаваме, че:

$$\begin{aligned}
\llbracket \tau_1 \rrbracket &= \llbracket \lambda\, \mathtt{x} : \mathtt{a}\, .\, \tau_1' \rrbracket \in [\llbracket \mathtt{a} \rrbracket \xrightarrow{\text{н}} \llbracket \mathtt{b} \rrbracket] \\
\llbracket \tau_1'[\mathtt{x}/\tau_2] \rrbracket &= \llbracket \mathtt{v} \rrbracket \in \llbracket \mathtt{b} \rrbracket.
\end{aligned}$$

Обядиняваме всичко и получаваме равенствата:

$$\llbracket \tau_1\tau_2 \rrbracket = \mathtt{eval}(\llbracket \tau_1 \rrbracket, \llbracket \tau_2 \rrbracket) \qquad \text{// от деф.}$$

$$
\begin{aligned}
&= [\![\tau_1]\!]([\![\tau_2]\!]) && // \ [\![\tau_1]\!] \in [[\![\mathtt{a}]\!] \xrightarrow{\text{н}} [\![\mathtt{b}]\!]] \\
&= [\![\lambda\, \mathtt{x} : \mathtt{a}\, .\, \tau_1']\!]([\![\tau_2]\!]) && // \ \textbf{(И.П.)} \\
&= [\![\tau_1']\!]_{\mathtt{x}:\mathtt{a}}([\![\tau_2]\!]) && \\
&= [\![\tau_1'[\mathtt{x}/\tau_2]]\!] && // \text{ от Лема за замяната} \\
&= [\![\mathtt{v}]\!] && // \ \textbf{(И.П.)}
\end{aligned}
$$

- Нека $\tau \equiv \mathtt{fix}(\tau')$. Тогава от правилата на операционната семантика имаме, че:

$$\frac{\tau'\ \mathtt{fix}(\tau') \Downarrow_{\mathtt{a}}^{\ell-1} \mathtt{v}}{\mathtt{fix}(\tau') \Downarrow_{\mathtt{a}}^{\ell} \mathtt{v}}\ \text{(fix)}$$

Тогава от **(И.П.)** имаме, че:

$$[\![\tau'\ \mathtt{fix}(\tau')]\!] = [\![\mathtt{v}]\!].$$

От правилата за типизиране знаем, че $\tau' : \mathtt{a} \to \mathtt{a}$. Сега остава да съобразим, че щом $[\![\tau']\!] \in [[\![\mathtt{a}]\!] \xrightarrow{\text{н}} [\![\mathtt{a}]\!]]$, то изображението $[\![\tau']\!]$ притежава най-малка неподвижна точка $\mathtt{lfp}([\![\tau']\!])$. Знаем, че според дефиницията на неподвижна точка, $[\![\tau']\!](\mathtt{lfp}([\![\tau']\!]) = \mathtt{lfp}([\![\tau']\!])$. Сега сме готови да завършим доказателството:

$$
\begin{aligned}
[\![\tau]\!] &= [\![\mathtt{fix}(\tau')]\!] && \\
&= \mathtt{lfp}([\![\tau']\!]) && // \text{ от деф.} \\
&= [\![\tau']\!](\mathtt{lfp}([\![\tau']\!])) && // \text{ неподв. точка} \\
&= [\![\tau']\!]([\![\mathtt{fix}(\tau')]\!]) && // \text{ от деф.} \\
&= \mathtt{eval}([\![\tau']\!], [\![\mathtt{fix}(\tau')]\!]) && \\
&= [\![\tau' \mathtt{fix}(\tau')]\!] && // \text{ от деф.} \\
&= [\![\mathtt{v}]\!]. && // \ \textbf{(И.П.)}
\end{aligned}
$$

□

Теоремата за коректност частично потвърждава нашата интуиция, че за типа nat можем да си мислим за $\bot^{[\![\mathtt{nat}]\!]}$ като за изчисление, което никога не завършва.

Другата посока ще я получим след малко.

Следствие 4.2. Нека τ е затворен терм от тип nat. Тогава е изпълнена импликацията

$$[\![\tau]\!] = \bot^{[\![\mathtt{nat}]\!]} \implies \tau \not\Downarrow_{\mathtt{nat}}.$$

За жалост, тази наша интуиция се „губи”, когато се интересуваме от термове от по-висок от nat тип. Нека просто да разгледаме терма $\Omega_{\mathtt{a}}'$ от тип $\mathtt{a} \to \mathtt{a}$. Имаме, че $[\![\Omega_{\mathtt{a}}']\!] = \bot^{[\![\mathtt{a}\to\mathtt{a}]\!]}$, но $\Omega_{\mathtt{a}}' \Downarrow_{\mathtt{a}\to\mathtt{a}} \Omega_{\mathtt{a}}'$.

За дискусия по този въпрос вижте [4, стр. 213]. Това означава, че ако искаме денотационната семантика да кореспондира по-точно с операционната семантика, ние трябва да въведем нов елемент за $\bot$ за области на Скот съответстващи на типове по-високи от nat.

4.7 Адекватност

Нашата цел в този раздел е да докажем следната теорема.

На англ. Adequacy

Теорема 4.2 (Теорема за адекватност). За всеки затворен терм $\tau : \mathtt{nat}$ е изпълнена импликацията

$$[\![\tau]\!] = n \neq \bot^{[\![\mathtt{nat}]\!]} \implies \tau \Downarrow_{\mathtt{nat}} \mathtt{n}.$$

Тук n е число, а $\mathtt{n}$ е константа.

Оказва се, че доказателството на тази теорема не е леко. Ще започнем като дефинираме за всеки тип $\mathtt{a}$ релацията $\lhd_{\mathtt{a}} \subseteq [\![\mathtt{a}]\!] \times \mathtt{PCF}_{\mathtt{a}}$ с индукция по построението на типовете.

Съобразете, че теоремата за адекватност на практика гласи, че $[\![\tau]\!] \lhd_{\mathtt{nat}} \tau$.

Обикновено $\lhd_{\mathtt{a}}$ се нарича *логическа релация* или formal approximation relation [6, стр. 57]. В [4, стр. 210] е обяснено защо имаме нужда от тези релации за да докажем теоремата за адекватност. В [7, стр. 134] е представен синтактичен подход към решаването на този проблем.

- Нека $\mathtt{a} = \mathtt{nat}$. Тогава

$$n \lhd_{\mathtt{nat}} \tau \stackrel{\text{деф}}{\iff} (n \neq \bot^{[\![\mathtt{nat}]\!]} \implies \tau \Downarrow_{\mathtt{nat}} \mathtt{n}).$$

- Нека $\mathtt{a} = \mathtt{b} \to \mathtt{c}$. Тогава

$$f \lhd_{\mathtt{b}\to\mathtt{c}} \tau \stackrel{\text{деф}}{\iff} (\forall e \in [\![\mathtt{b}]\!])(\forall \mu \in \mathtt{PCF}_{\mathtt{b}})[\, e \lhd_{\mathtt{b}} \mu \implies f(e) \lhd_{\mathtt{c}} \tau(\mu)\,].$$

- Нека разгледаме типовия контекст $\Gamma = \mathtt{x}_1 : \mathtt{a}_1, \dots, \mathtt{x}_n : \mathtt{a}_n$. Тогава

$$(u_1, \dots, u_n) \lhd_{\Gamma} (\tau_1, \dots, \tau_n) \stackrel{\text{деф}}{\iff} u_1 \lhd_{\mathtt{a}_1} \tau_1 \;\&\; \cdots \;\&\; u_n \lhd_{\mathtt{a}_n} \tau_n.$$

Пример 4.3. Да проверим внимателно защо е изпълнено, че:

$$\mathtt{id}_{[\![\mathtt{nat}]\!]} \lhd_{\mathtt{nat}\to\mathtt{nat}} \lambda\, \mathtt{x} : \mathtt{nat}\, .\, \mathtt{x}\ +\ \mathtt{0}.$$

Според дефиницията трябва да проверим импликацията

$$e \lhd_{\mathtt{nat}} \mu \implies \mathtt{id}_{[\![\mathtt{nat}]\!]}(e) \lhd_{\mathtt{nat}} \tau(\mu),$$

за произволен елемент $e \in \mathbb{N}_\bot$ и произволен затворен терм $\mu : \mathtt{nat}$.

Аналогично можем да видим, че $\mathtt{id}_{[\![\mathtt{nat}]\!]} \lhd_{\mathtt{nat}} \lambda\ \mathtt{x} : \mathtt{nat}\, .\, x$.

- Ако $e = \bot$, то от дефиницията на $\lhd_{\mathtt{nat}}$ веднага следва, че за произволен затворен терм $\mu : \mathtt{nat}$, то $\bot \lhd_{\mathtt{nat}} \mu$. Понеже $\mathtt{id}_{[\![\mathtt{nat}]\!]}(\bot) = \bot$, то отново от дефиницията веднага следва, че $\bot \lhd_{\mathtt{nat}} \tau(\mu)$, за произволен затворен терм $\mu : \mathtt{nat}$.

- Нека $e = n \in \mathbb{N}$ и да разгледаме затворен терм $\mu : \mathtt{nat}$, за който $n \lhd_{\mathtt{nat}} \mu$. Според дефиницията на $\lhd_{\mathtt{nat}}$, това означава, че $\mu \Downarrow_{\mathtt{nat}} \mathtt{n}$. Сега да видим защо $\mathtt{id}_{[\![\mathtt{nat}]\!]}(n) = n \lhd_{\mathtt{nat}} \tau(\mu)$ или с други думи, трябва да проверим, че $\tau(\mu) \Downarrow_{\mathtt{nat}} \mathtt{n}$. Тук се позоваваме на правилата от операционната семантика:

$$\dfrac{(\text{val})\;\dfrac{\tau \text{ е стойност}}{\tau \Downarrow_{\mathtt{nat}\to\mathtt{nat}} \lambda\, \mathtt{x} : \mathtt{nat}\, .\, \mathtt{x}\ +\ \mathtt{0}} \qquad \dfrac{\dfrac{n \lhd_{\mathtt{nat}} \mu}{\mu \Downarrow_{\mathtt{nat}} \mathtt{n}} \qquad \dfrac{\mathtt{0} \text{ е стойност}}{\mathtt{0} \Downarrow_{\mathtt{nat}} \mathtt{0}}\;(\text{val})}{(\mathtt{x+0})\,[\mathtt{x}/\mu] \Downarrow_{\mathtt{nat}} \mathtt{n}}\;(\text{plus})}{\tau(\mu) \Downarrow_{\mathtt{nat}} \mathtt{n}}\;(\text{app})$$

Нека първо да разгледаме някои основни свойства на релацията $\lhd_{\mathtt{a}}$. Тук доказателствата протичат с индукция по построението на типовете. [7, стр. 197]

Твърдение 4.6. За всеки тип $\mathtt{a}$ и всеки затворен терм $\tau : \mathtt{a}$ е изпълнено, че $\bot^{[\![\mathtt{a}]\!]} \lhd_{\mathtt{a}} \tau$.

Доказателство. Индукция по построението на типовете $\mathtt{a}$. Първо, нека $\mathtt{a} = \mathtt{nat}$. По тривиални съображения имаме, че за произволен терм $\tau : \mathtt{a}$ е изпълнено, че $\bot^{[\![\mathtt{nat}]\!]} \lhd_{\mathtt{nat}} \tau$.

Второ, нека $\mathtt{a} = \mathtt{b} \to \mathtt{c}$ и да фиксираме произволен терм $\tau : \mathtt{b} \to \mathtt{c}$. Тук имаме, че $\bot^{[\![\mathtt{a}]\!]} \in [[\![\mathtt{b}]\!] \xrightarrow{\text{н}} [\![\mathtt{c}]\!]]$ е изображение, за което $\bot^{[\![\mathtt{a}]\!]}(e) = \bot^{[\![\mathtt{c}]\!]}$ за всеки елемент $e \in [\![\mathtt{b}]\!]$. Нека $e \lhd_{\mathtt{b}} \mu$, където $\mu : \mathtt{b}$. Щом $\tau : \mathtt{b} \to \mathtt{c}$, от правилата за типизиране е ясно, че $\tau(\mu) : \mathtt{c}$. Сега от **(И.П.)** за типа $\mathtt{c}$ е ясно, че $\bot^{[\![\mathtt{a}]\!]}(e) = \bot^{[\![\mathtt{c}]\!]} \lhd_{\mathtt{c}} \tau(\mu)$. □

Задача 4.3. Нека положим $\tau \equiv \lambda\, \mathtt{x} : \mathtt{nat}\, .\, \lambda\, \mathtt{y} : \mathtt{nat}\, .\, x - y$. Проверете, че

$$f \lhd_{\mathtt{nat}\to\mathtt{nat}\to\mathtt{nat}} \tau,$$

където:

- $f = \mathtt{curry((-))}$;
- $f(a)(b) = \begin{cases} a - b, & \text{ако } a \geq b \\ \bot, & \text{иначе} \end{cases}$;
- $f(a)(b) = \bot$ за произволни $a, b \in \mathbb{N}_\bot$.

Тази задача ни подсказва следното свойство.

Твърдение 4.7. За произволен тип a, нека $f \lhd_{\mathtt{a}} \tau$. Ако $g \sqsubseteq f$, то $g \lhd_{\mathtt{a}} \tau$.

Доказателство. Индукция по типовете. □

Твърдение 4.8. Нека за произволен тип $\mathtt{a}$ и произволен затворен терм $\tau : \mathtt{a}$ да разгледаме множеството $D \stackrel{\text{деф}}{=} \{d \in [\![\mathtt{a}]\!] \mid d \lhd_{\mathtt{a}} \tau\}$. Тогава ако $(d_i)_{i=0}^{\infty}$ е верига от елементи на D, то $\bigsqcup_i d_i$ също принадлежи на D.

Доказателство. Индукция по построението на типовете $\mathtt{a}$. Първо, нека $\mathtt{a} = \mathtt{nat}$. Нека $(d_i)_{i=0}^{\infty}$ е верига от елементи на $\mathbb{N}_\perp$ и за всеки индекс i, $d_i \lhd_{\mathtt{nat}} \tau$. Ако за всяко i, $d_i = \perp$, то $\bigsqcup_i d_i = \perp^{[\![\mathtt{nat}]\!]}$ и следователно $\bigsqcup_i d_i \lhd_{\mathtt{nat}} \tau$. Ако съществува индекс i_0, за който $d_{i_0} = n \neq \perp$, то е ясно, че за всяко $i > i_0$, $d_i = n$. Оттук следва, че $\bigsqcup_i d_i = n = d_{i_0}$. Понеже $d_{i_0} \lhd_{\mathtt{nat}} \tau$, то директно следва, че $\bigsqcup_i d_i \lhd_{\mathtt{nat}} \tau$.

Да напомним, че $[\![\mathtt{nat}]\!] = \mathbb{N}_\perp$. Ясно е, че всяка верига от елементи на $\mathbb{N}_\perp$ се стабилизира.

Второ, нека $\mathtt{a} = \mathtt{b} \to \mathtt{c}$ и да фиксираме произволен терм $\tau : \mathtt{b} \to \mathtt{c}$. Нека $(f_i)_{i=0}^{\infty}$ е верига от елементи на $[[\![\mathtt{b}]\!] \stackrel{\text{н}}{\to} [\![\mathtt{c}]\!]]$, за които е изпълнено, че $f_i \lhd_{\mathtt{a}} \tau$. Трябва да докажем, че $\bigsqcup_i f_i \lhd_{\mathtt{a}} \tau$, т.е. за произволен елемент $e \in [\![\mathtt{b}]\!]$ и произволен затворен терм $\mu : \mathtt{b}$, за който $e \lhd_{\mathtt{b}} \mu$, то $(\bigsqcup_i f)(e) \lhd_{\mathtt{c}} \tau(\mu)$. Но ние знаем от *Лема* 1.2, че $(\bigsqcup_i f)(e) = \bigsqcup_i f_i(e)$. Щом $f_i \lhd_{\mathtt{b}\to\mathtt{c}} \tau$, то за разглежданите e и μ имаме, че $f_i(e) \lhd_{\mathtt{c}} \tau(\mu)$. Ние знаем, че $(f_i(e))_{i=0}^{\infty}$ е верига и от **(И.П.)** за типа $\mathtt{c}$ следва, че $\bigsqcup_i \{f_i(e)\} \lhd_{\mathtt{c}} \tau(\mu)$. □

Твърдение 4.9. Да разгледаме произволен тип $\mathtt{a}$ и произволен затворен терм $\tau : \mathtt{a}$. Тогава е изпълнено следното:

$$\frac{d \lhd_{\mathtt{a}} \tau \qquad \tau \leq_{op} \rho : \mathtt{a}}{d \lhd_{\mathtt{a}} \rho}$$

Доказателство. Индукция по построението на типовете $\mathtt{a}$. Първо, нека $\mathtt{a} = \mathtt{nat}$. Нека $d \lhd_{\mathtt{nat}} \tau$ и $(\forall \mathtt{v} \in \mathtt{Value}_{\mathtt{nat}})[\tau \Downarrow_{\mathtt{nat}} \mathtt{v} \implies \rho \Downarrow_{\mathtt{nat}} \mathtt{v}]$. Понеже $\sqsubseteq$ е плоската наредба в $\mathbb{N}_\perp$, то имаме два случая. Ако $d = \perp^{[\![\mathtt{nat}]\!]}$, то е ясно от *Твърдение* 4.6, че $d \lhd_{\mathtt{nat}} \rho$. Нека сега $d \neq \perp^{[\![\mathtt{nat}]\!]}$. Понеже $d \lhd_{\mathtt{nat}} \tau$, то $\tau \Downarrow_{\mathtt{nat}} \mathtt{d}$. Оттук следва, че $\rho \Downarrow_{\mathtt{nat}} \mathtt{d}$. Заключаваме, че $d \lhd_{\mathtt{nat}} \rho$.

Да напомним, че с $\mathtt{v}$ означаваме термове стойности.

Второ, нека $\mathtt{a} = \mathtt{b} \to \mathtt{c}$ и да фиксираме произволен терм $\tau : \mathtt{b} \to \mathtt{c}$. Нека

$$f \lhd_{\mathtt{b}\to\mathtt{c}} \tau \tag{4.1}$$

$$(\forall \mathtt{v} \in \mathtt{Value}_{\mathtt{b}\to\mathtt{c}})[\tau \Downarrow_{\mathtt{b}\to\mathtt{c}} \mathtt{v} \implies \rho \Downarrow_{\mathtt{b}\to\mathtt{c}} \mathtt{v}] \tag{4.2}$$

Ще докажем, че $f \lhd_{\mathtt{b}\to\mathtt{c}} \rho$. За целта, да разгледаме произволен елемент $e \in [\![\mathtt{b}]\!]$ и произволен затворен терм $\mu : \mathtt{b}$, за който $e \lhd_{\mathtt{b}} \mu$. Достатъчно е да докажем, че $f(e) \lhd_{\mathtt{c}} \rho(\mu)$. За момента от Свойство (4.1) знаем само, че $f(e) \lhd_{\mathtt{c}} \tau(\mu)$. Понеже имаме следното правило в операционната семантика:

$$\frac{\tau \Downarrow_{\mathtt{b}\to\mathtt{c}} \overbrace{\lambda\, \mathtt{x} : \mathtt{b} \,.\, \tau'}^{\mathtt{v}} \qquad \tau'[\mathtt{x}/\mu] \Downarrow_{\mathtt{c}} \mathtt{v}'}{\tau(\mu) \Downarrow_{\mathtt{c}} \mathtt{v}'}\ (\text{app})$$

Имаме от (4.2), че

$$\tau \Downarrow_{\mathtt{b}\to\mathtt{c}} \mathtt{v} \implies \rho \Downarrow_{\mathtt{b}\to\mathtt{c}} \mathtt{v},$$

а тук $\mathtt{v} \equiv \lambda\, \mathtt{x} : \mathtt{b} \,.\, \tau'$.

то от Свойство (4.2) получаваме, че

$$\frac{\rho \Downarrow_{\mathtt{b}\to\mathtt{c}} \overbrace{\lambda\, \mathtt{x} : \mathtt{b} \,.\, \tau'}^{\mathtt{v}} \qquad \tau'[\mathtt{x}/\mu] \Downarrow_{\mathtt{c}} \mathtt{v}'}{\rho(\mu) \Downarrow_{\mathtt{c}} \mathtt{v}'}\ (\text{app})$$

Оттук следва, че

$$(\forall \mathtt{v}' \in \mathtt{Value}_{\mathtt{c}})[\tau(\mu) \Downarrow_{\mathtt{c}} \mathtt{v}' \implies \rho(\mu) \Downarrow_{\mathtt{c}} \mathtt{v}'],$$

или с други думи,

$$\tau(\mu) \leq_{op} \rho(\mu) : \mathtt{c}. \tag{4.3}$$

Сега от **(И.П.)** за типа c директно следва, че щом $f(e) \lhd_{\mathtt{c}} \tau(\mu)$ и Свойство (4.3), то

$$f(e) \lhd_{\mathtt{c}} \rho(\mu).$$

□

За да докажем, че $[\![\tau]\!] \lhd_{\mathtt{nat}} \tau$, то трябва да докажем, че за всеки тип a и всеки затворен терм τ от тип a, че е изпълнено $[\![\tau]\!] \lhd_{\mathtt{a}} \tau$ с индукция по построението на термовете. Тук обаче имаме проблем. Ако $\tau \equiv \lambda\, \mathtt{y} : \mathtt{b} \,.\, \tau_1$, то трябва да използваме индукционно предположение за τ_1, в който вече има свободна променлива y. Поради тази причина, ние трябва да разгледаме едно по-общо твърдение, при което позволяваме в термовете да се срещат свободни променливи.

Лема 4.5 (Фундаментално свойство на $\lhd_{\mathtt{a}}$)**.** Нека $\Gamma = \mathtt{x}_1 : \mathtt{a}_1, \ldots, \mathtt{x}_n : \mathtt{a}_n$. Тогава

$$\frac{\Gamma \vdash \tau : \mathtt{a} \qquad (u_1, \ldots, u_n) \lhd_{\Gamma} (\mu_1, \ldots, \mu_n)}{[\![\tau]\!]_{\Gamma}(\overline{u}) \lhd_{\mathtt{a}} \tau[\overline{\mathtt{x}}/\overline{\mu}]}$$

Доказателство. Индукция по построението на термовете.

- Нека $\tau \equiv \mathtt{n}$. Тук директно от дефиницията на релацията $\lhd_{\mathtt{nat}}$ имаме, че

$$n \lhd_{\mathtt{nat}} \mathtt{n}.$$

- Нека $\tau \equiv \mathtt{x}_i$. Този случай също е много лесен. Имаме, че $\tau[\overline{x}/\overline{\mu}] \equiv \mu_i$ и $[\![\tau]\!]_{\Gamma}(\overline{u}) = u_i$. Тогава, понеже $(u_1, \ldots, u_n) \lhd_{\Gamma} (\mu_1, \ldots, \mu_n)$, то директно получаваме, че

$$[\![\tau]\!]_{\Gamma}(\overline{u}) \lhd_{\mathtt{a}} \tau[\overline{x}/\overline{\mu}].$$

Понеже $\Gamma \vdash \tau : \mathtt{a}$, то няма как τ да е променлива, която да не е някоя измежду $\mathtt{x}_1, \ldots, \mathtt{x}_n$.

- Нека $\tau \equiv \tau_1 + \tau_2$. Тогава от правилата за типизиране имаме, че $\mathtt{a} = \mathtt{nat}$. Имаме също, че

$$\tau[\overline{x}/\overline{\mu}] \equiv \tau_1[\overline{x}/\overline{\mu}] + \tau_2[\overline{x}/\overline{\mu}];$$
$$[\![\tau]\!]_\Gamma(\overline{u}) = (\mathtt{+})(\underbrace{[\![\tau_1]\!]_\Gamma(\overline{u})}_{n_1}, \underbrace{[\![\tau_2]\!]_\Gamma(\overline{u})}_{n_2}) = n.$$

  Можем да приемем, че $n \neq \bot$, защото иначе този случай е тривиален заради дефиницията на $\lhd_{\mathtt{nat}}$. От **(И.П.)** имаме, че $[\![\tau_1]\!]_\Gamma(\overline{u}) \lhd_{\mathtt{nat}} \tau_1[\overline{x}/\overline{\mu}]$ и $[\![\tau_2]\!]_\Gamma(\overline{u}) \lhd_{\mathtt{nat}} \tau_2[\overline{x}/\overline{\mu}]$. Това означава, че $\tau_1[\overline{x}/\overline{\mu}] \Downarrow_{\mathtt{nat}} \mathtt{n}_1$ и $\tau_2[\overline{x}/\overline{\mu}] \Downarrow_{\mathtt{nat}} \mathtt{n}_2$. От правилата на операционната семантика е ясно, че $\tau[\overline{x}/\overline{\mu}] \Downarrow_{\mathtt{nat}} \mathtt{n}$ и следователно

$$[\![\tau]\!]_\Gamma(\overline{u}) \lhd_{\mathtt{nat}} \tau[\overline{\mathtt{x}}/\overline{\mu}].$$

- Нека $\tau \equiv \tau_1 \mathtt{\ -\ } \tau_2$.

- Нека $\tau \equiv \tau_1 \mathtt{\ ==\ } \tau_2$.

- Нека $\tau \equiv \mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3$. От правилата за типизиране имаме, че $\Gamma \vdash \tau_1 : \mathtt{nat}$ и $\Gamma \vdash \tau_2 : \mathtt{a}$ и $\Gamma \vdash \tau_3 : \mathtt{a}$. Знаем също, че

$$\tau[\overline{x}/\overline{\mu}] \equiv \mathtt{if}\ \tau_1[\overline{x}/\overline{\mu}]\ \mathtt{then}\ \tau_2[\overline{x}/\overline{\mu}]\ \mathtt{else}\ \tau_3[\overline{x}/\overline{\mu}]$$
$$[\![\tau]\!]_\Gamma(\overline{u}) = \mathtt{if}([\![\tau_1]\!]_\Gamma(\overline{u}), [\![\tau_2]\!]_\Gamma(\overline{u}), [\![\tau_3]\!]_\Gamma(\overline{u})).$$

  Случаят, когато $[\![\tau_1]\!]_\Gamma(\overline{u}) = \bot^{[\![\mathtt{nat}]\!]}$ е очевиден и ще го пропуснем. Нека да разгледаме случая, когато $[\![\tau_1]\!]_\Gamma(\overline{u}) = 0$. Тогава $[\![\tau]\!]_\Gamma(\overline{u}) = [\![\tau_3]\!]_\Gamma(\overline{u})$. От **(И.П.)** за τ_1 получаваме, че

  Случаят, когато $[\![\tau_1]\!]_\Gamma(\overline{u}) > 0$ е сходен и ще го оставим за трудолюбивия читател.

$$[\![\tau_1]\!]_\Gamma(\overline{u}) \lhd_{\mathtt{nat}} \tau_1[\overline{x}/\overline{\mu}].$$

  От дефиницията на $\lhd_{\mathtt{nat}}$ следва, че щом $[\![\tau_1]\!]_\Gamma(\overline{u}) = 0$, то $\tau_1[\overline{x}/\overline{\mu}] \Downarrow_{\mathtt{nat}} \mathtt{0}$. От **(И.П.)** за τ_3 получаваме, че

$$\underbrace{[\![\tau_3]\!]_\Gamma(\overline{u})}_{[\![\tau]\!]_\Gamma(\overline{u})} \lhd_{\mathtt{a}} \tau_3[\overline{x}/\overline{\mu}].$$

  Сега от правилата на операционната семантика имаме, че

$$\frac{\tau_1[\overline{x}/\overline{\mu}] \Downarrow_{\mathtt{nat}} \mathtt{0} \qquad \tau_3[\overline{x}/\overline{\mu}] \Downarrow_{\mathtt{a}} \mathtt{v}}{\underbrace{\mathtt{if}\ \tau_1[\overline{x}/\overline{\mu}]\ \mathtt{then}\ \tau_2[\overline{x}/\overline{\mu}]\ \mathtt{else}\ \tau_3[\overline{x}/\overline{\mu}]}_{\tau[\overline{x}/\overline{\mu}]} \Downarrow_{\mathtt{a}} \mathtt{v}}\ (\mathrm{if}_0)$$

  Да напишем на чисто важните неща, които имаме до момента.

  - $[\![\tau]\!]_\Gamma(\overline{u}) \lhd_{\mathtt{a}} \tau_3[\overline{x}/\overline{\mu}]$;
  - $(\forall \mathtt{v} \in \mathtt{Value}_{\mathtt{a}})[\tau_3[\overline{x}/\overline{\mu}] \Downarrow_{\mathtt{a}} \mathtt{v} \implies \tau[\overline{x}/\overline{\mu}] \Downarrow_{\mathtt{a}} \mathtt{v}]$.

  Сега директно прилагаме *Твърдение* 4.9 и получаваме, че $[\![\tau]\!]_\Gamma(\overline{u}) \lhd_{\mathtt{a}} \tau[\overline{x}/\overline{\mu}]$.

- Нека $\tau \equiv \tau_1\tau_2$. От правилата за типизиране имаме, че

$$\frac{\Gamma \vdash \tau_1 : \mathtt{b} \to \mathtt{a} \qquad \Gamma \vdash \tau_2 : \mathtt{b}}{\Gamma \vdash \tau_1\tau_2 : \mathtt{a}}$$

Да напомним, че

$$[\![\tau_1\tau_2]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} \mathtt{eval}([\![\tau_1]\!]_\Gamma(\overline{u}), [\![\tau_2]\!]_\Gamma(\overline{u})).$$

Понеже τ е съставен от τ_1 и τ_2, от **(И.П.)** имаме следното:

$$\begin{aligned}
&[\![\tau_1]\!]_\Gamma(\overline{u}) \lhd_{\mathtt{b}\to\mathtt{a}} \tau_1[\overline{x}/\overline{\mu}];\\
&[\![\tau_2]\!]_\Gamma(\overline{u}) \lhd_{\mathtt{b}} \tau_2[\overline{x}/\overline{\mu}].
\end{aligned}$$

Щом $[\![\tau_1]\!]_\Gamma(\overline{u}) \lhd_{\mathtt{b}\to\mathtt{a}} \tau_1[\overline{x}/\overline{\mu}]$, то от дефиницията на релацията $\lhd_{\mathtt{b}\to\mathtt{a}}$ следва, че за произволни $e \lhd_{\mathtt{b}} \rho$ имаме, че $\mathtt{eval}([\![\tau_1]\!]_\Gamma(\overline{u}), e) \lhd_{\mathtt{a}} \tau_1[\overline{x}/\overline{\mu}](\rho)$. Нека сега да вземем $e \stackrel{\text{деф}}{=} [\![\tau_2]\!]_\Gamma(\overline{u})$ и $\rho \stackrel{\text{деф}}{=} \tau_2[\overline{x}/\overline{\mu}]$. Така получаваме

$$\underbrace{\mathtt{eval}([\![\tau_1]\!]_\Gamma(\overline{u}), [\![\tau_2]\!]_\Gamma(\overline{u}))}_{[\![\tau]\!]_\Gamma(\overline{u})} \lhd_{\mathtt{a}} \underbrace{\tau_1[\overline{x}/\overline{\mu}](\tau_2[\overline{x}/\overline{\mu}])}_{\tau[\overline{x}/\overline{\mu}]}.$$

- Нека $\tau = \lambda\, \mathtt{y} : \mathtt{b} \,.\, \tau'$. Тогава от правилата за типизиране следва, че $\mathtt{a} = \mathtt{b} \to \mathtt{c}$, за някои типове $\mathtt{b}$ и $\mathtt{c}$, и $\Gamma, \mathtt{y} : \mathtt{b} \vdash \tau' : \mathtt{c}$. Да напомним, че

$$[\![\tau]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} \mathtt{curry}([\![\tau']\!]_{\Gamma,\mathtt{y}:\mathtt{b}})(\overline{u}) \in [[\![\mathtt{b}]\!] \stackrel{\text{н}}{\to} [\![\mathtt{c}]\!]].$$

Да положим $f \stackrel{\text{деф}}{=} [\![\tau]\!]_\Gamma(\overline{u})$. Тогава лесно се съобразява, че:

$$\begin{aligned}
f(e) &= [\![\tau]\!]_\Gamma(\overline{u})(e)\\
&= \mathtt{curry}([\![\tau']\!]_{\Gamma,\mathtt{y}:\mathtt{b}})(\overline{u})(e)\\
&= [\![\tau']\!]_{\Gamma,\mathtt{y}:\mathtt{b}}(\overline{u}, e).
\end{aligned}$$

Трябва да докажем, че $f \lhd_{\mathtt{b}\to\mathtt{c}} \tau[\overline{x}/\overline{\mu}]$. Това означава, че за произволни $e \in [\![\mathtt{b}]\!]$ и $\rho : \mathtt{b}$, за които $e \lhd_{\mathtt{b}} \rho$, трябва да докажем, че $f(e) \lhd_{\mathtt{c}} \tau[\overline{x}/\overline{\mu}](\rho)$.

Имаме, че

$$\frac{\Gamma, \mathtt{y} : \mathtt{b} \vdash \tau' : \mathtt{c} \qquad (u_1, \ldots, u_n, e) \lhd_{\Gamma,\mathtt{y}:\mathtt{b}} (\mu_1, \ldots, \mu_n, \rho)}{\underbrace{[\![\tau']\!]_{\Gamma,\mathtt{y}:\mathtt{b}}(\overline{u}, e)}_{f(e)} \lhd_{\mathtt{c}} \tau'[\overline{x}/\overline{\mu}][y/\rho]}\ \textbf{(И.П.)}$$

От правилата на операционната семантика имаме следното:

$$\frac{\dfrac{\lambda\, \mathtt{y} : \mathtt{b} \,.\, \tau'[\overline{x}/\overline{\mu}] \in \mathtt{Value}_{\mathtt{b}\to\mathtt{c}}}{\lambda\, \mathtt{y} : \mathtt{b} \,.\, \tau'[\overline{x}/\overline{\mu}] \Downarrow_{\mathtt{b}\to\mathtt{c}} \lambda\, \mathtt{y} : \mathtt{b} \,.\, \tau'[\overline{x}/\overline{\mu}]} \qquad \tau'[\overline{x}/\overline{\mu}][y/\rho] \Downarrow_{\mathtt{c}} \mathtt{v}}{(\underbrace{\lambda\, \mathtt{y} : \mathtt{b} \,.\, \tau'[\overline{x}/\overline{\mu}]}_{\tau[\overline{x}/\overline{\mu}]})(\rho) \Downarrow_{\mathtt{c}} \mathtt{v}}\ \text{(cbn)}$$

Да обобщим какво имаме до момента:

- $f(e) \lhd_{\mathtt{c}} \tau'[\overline{x}/\overline{\mu}][y/\rho]$;
- $(\forall \mathtt{v} \in \mathtt{Value}_{\mathtt{c}})[\, \tau'[\overline{x}/\overline{\mu}][y/\rho] \Downarrow_{\mathtt{c}} \mathtt{v} \implies \tau[\overline{x}/\overline{\mu}] \Downarrow_{\mathtt{c}} \mathtt{v}\,]$.

От *Твърдение* 4.9 веднага заключаваме, че $f(e) \lhd_{\mathtt{c}} \tau[\overline{x}/\overline{\mu}](\rho)$, което трябваше да докажем.

- Нека $\tau \equiv \mathtt{fix}(\tau')$. Тук трябва да докажем, че $[\![\mathtt{fix}(\tau')]\!](\overline{u}) \lhd_{\mathtt{a}} \tau[\overline{x}/\overline{\mu}]$. От правилата за типизиране имаме, че $\Gamma \vdash \tau' : \mathtt{a} \to \mathtt{a}$. От **(И.П.)** , приложено за τ', имаме, че
$$[\![\tau']\!]_\Gamma(\overline{u}) \lhd_{\mathtt{a}\to\mathtt{a}} \tau'[\overline{x}/\overline{\mu}].$$
Нека за улеснение да положим $f \stackrel{\text{деф}}{=} [\![\tau']\!]_\Gamma(\overline{u}) \in [[\![\mathtt{a}]\!] \stackrel{\text{н}}{\to} [\![\mathtt{a}]\!]]$. Да напомним, че
$$[\![\mathtt{fix}(\tau')]\!]_\Gamma(\overline{u}) = \mathtt{lfp}(f).$$
От *Твърдение* 4.8 знаем, че
$$D \stackrel{\text{деф}}{=} \{d \in [\![\mathtt{a}]\!] \mid d \lhd_{\mathtt{a}} \mathtt{fix}(\tau'[\overline{x}/\overline{\mu}])\}$$
е непрекъснато свойство. Целта ни е да докажем, че $\mathtt{lfp}(f) \in D$. Ще приложим правилото на Скот.

  От *Твърдение* 4.6 имаме, че $\bot^{[\![\mathtt{a}]\!]} \in D$. Нека сега $d \in D$. Ще докажем, че $f(d) \in D$.

  Понеже $f \lhd_{\mathtt{a}\to\mathtt{a}} \tau'[\overline{x}/\overline{\mu}]$, то за произволни e и ρ, за които $e \lhd_{\mathtt{a}} \rho$, е изпълнено, че $f(e) \lhd_{\mathtt{a}} \tau'[\overline{x}/\overline{\mu}](\rho)$. Нека изберем $\rho \stackrel{\text{деф}}{=} \mathtt{fix}(\tau'[\overline{x}/\overline{\mu}])$. Щом $d \in D$, то е ясно, че $d \lhd_{\mathtt{a}} \rho$ и следователно
$$f(d) \lhd_{\mathtt{a}} \tau'[\overline{x}/\overline{\mu}](\underbrace{\mathtt{fix}(\tau'[\overline{x}/\overline{\mu}])}_{\rho}).$$
От правилата на операционната семантика имаме, че:
$$\frac{\tau'[\overline{x}/\overline{\mu}](\mathtt{fix}(\tau'[\overline{x}/\overline{\mu}])) \Downarrow_{\mathtt{a}} \mathtt{v}}{\mathtt{fix}(\tau'[\overline{x}/\overline{\mu}]) \Downarrow_{\mathtt{a}} \mathtt{v}}\ (\text{fix})$$
Тогава от *Твърдение* 4.9 следва, че:
$$f(d) \lhd_{\mathtt{a}} \mathtt{fix}(\tau'[\overline{x}/\overline{\mu}]).$$
Така доказахме, че $f(d) \in D$. Заключаваме, че $\mathtt{lfp}(f) \in D$, т.е.
$$[\![\mathtt{fix}(\tau')]\!]_\Gamma(\overline{u}) \lhd_{\mathtt{a}} \mathtt{fix}(\tau'[\overline{x}/\overline{\mu}])$$

$\square$

Следствие 4.3. За всеки тип $\mathtt{a}$ и за всеки затворен терм τ е изпълнено свойството:

$$\tau \in \mathrm{PCF}_{\mathtt{a}} \implies [\![\tau]\!] \lhd_{\mathtt{a}} \tau.$$

Така на практика доказахме теоремата за адекватност.

Теорема 4.3 (Теорема за адекватност)**.** За всеки затворен терм $\tau : \mathtt{nat}$,

$$[\![\tau]\!] = n \neq \bot^{[\![\mathtt{nat}]\!]} \implies \tau \Downarrow_{\mathtt{nat}} \mathtt{n}.$$

Доказателство. Да разгледаме произволен затворен терм $\tau : \mathtt{nat}$. Нека $[\![\tau]\!] = n \neq \bot$. От *Следствие* 4.3 имаме, че $[\![\tau]\!] \lhd_{\mathtt{nat}} \tau$. Тогава от дефиницията на $\lhd_{\mathtt{nat}}$ получаваме, че $\tau \Downarrow_{\mathtt{nat}} \mathtt{n}$. □

Следствие 4.4. За всеки затворен терм $\tau : \mathtt{nat}$ е изпълнено, че:

$$[\![\tau]\!] = n \neq \bot^{[\![\mathtt{nat}]\!]} \text{ точно тогава, когато } \tau \Downarrow_{\mathtt{nat}} \mathtt{n}.$$

Доказателство. Посоката $(\Rightarrow)$ е теоремата за адекватност. Посоката $(\Leftarrow)$ е теоремата за коректност. □

Да разгледаме термовете

$$\tau \equiv \lambda\, \mathtt{x} : \mathtt{nat}\, .\, \mathtt{x\ +\ 0}$$
$$\rho \equiv \lambda\, \mathtt{x} : \mathtt{nat}\, .\, \mathtt{x}.$$

Ясно е, че $[\![\tau]\!] = [\![\rho]\!]$, но според правилата на операционната семантика, понеже τ е стойност, то $\tau \not\Downarrow_{\mathtt{nat}\to\mathtt{nat}} \rho$. Оттук веднага е ясно, че няма как да имаме теорема за адекватност за по-високи типове от $\mathtt{nat}$.

Следващото следствие ни казва, че за типа $\mathtt{nat}$, $\bot^{[\![\mathtt{nat}]\!]}$ означава изчисление, което никога не завършва.

Следствие 4.5. За всеки затворен терм $\tau : \mathtt{nat}$ е изпълнено, че:

$$[\![\tau]\!] = \bot^{[\![\mathtt{nat}]\!]} \text{ точно тогава, когато } \tau \not\Downarrow_{\mathtt{nat}}.$$

4.8 Контекстна еквивалентност

[7, стр. 177], [8, Глава 48]

Основен въпрос, който можем да зададем е кога две програми (затворени термове) имат едно и също значение.

Първо, можем да използваме операционната семантика. Можем да кажем, че два затворени терма τ_1 и τ_2 от тип a имат едно и също значение, ако $\tau_1 \cong_{op} \tau_2 : \mathtt{a}$. За съжаление, този подход не е особено удачен за типове по-сложни от nat. Да вземем $\tau_1 \stackrel{\text{деф}}{=} \lambda\, \mathtt{x} : \mathtt{nat}\, .\, 0$ и $\tau_2 \stackrel{\text{деф}}{=} \lambda\, \mathtt{x} : \mathtt{nat}\, .\, (\lambda\, \mathtt{y} : \mathtt{nat}\, .\, y)0$. Ясно е, че $\tau_1 \ncong_{op} \tau_2 : \mathtt{nat} \to \mathtt{nat}$, макар и τ_1 и τ_2 на практика да имат едно и също значение.

Второ, можем да използваме денотационната семантика за тази цел. Този подход е често срещан. Можем да кажем, че две програми τ_1 и τ_2 имат един и същ смисъл, ако $\tau_1 \cong_{den} \tau_2 : \mathtt{a}$.

Трето, възможно е да разгледаме и следната интерпретация на идеята за смисъл на една програма, която все пак използва операционната семантика. Може да кажем, че две програми τ_1 и τ_2 имат същото значение, ако за произволна по-голяма програма ρ, в която участва τ_1, ако заменим навсякъде τ_1 с τ_2 в ρ, то няма да променим значението ѝ. Поради направения по-горе коментар, ще изискваме да разглеждаме само програми ρ от тип nat.

Сега ще се опитваме да формализираме това трето предложение за дефиниция на понятието смисъл на програма. Ще започнем с дефиницията на **контекст**.

Интуитивно, контекстите са програмни фрагменти. Не са пълни програми, защото имат празни места означени с $-$.

$$\mathcal{C} ::= - \mid \mathtt{n} \mid \mathtt{x} \mid \mathcal{C} - \mathcal{C} \mid \mathcal{C} + \mathcal{C} \mid \mathcal{C} == \mathcal{C} \mid \mathtt{if}\ \mathcal{C}\ \mathtt{then}\ \mathcal{C}\ \mathtt{else}\ \mathcal{C} \mid \mathcal{C}\mathcal{C} \mid \lambda\, \mathtt{x} : \mathtt{a}\, .\, \mathcal{C} \mid \mathtt{fix}(\mathcal{C}).$$

Контекстите са на практика изрази, като позволяваме да имат специална свободна променлива, която означаваме с $-$. За краткост, вместо $\mathcal{C}[-/\tau]$ ще пишем $\mathcal{C}[\tau]$.

Ще казваме, че **завършена програма** е затворен терм от тип nat.

Някои наричат тази релация observational equivalence. Тук наричаме релацията contextual equivalence.

Определение 4.6. За всеки два затворени терма τ_1 и τ_2 и тип a дефинираме релацията

$$\tau_1 \leq_{ctx} \tau_2 : \mathtt{a}$$

по следния начин:

1) $\emptyset \vdash \tau_1 : \mathtt{a}$ и $\emptyset \vdash \tau_2 : \mathtt{a}$

2) За *всички* контексти $\mathcal{C}[-]$, за които $\emptyset \vdash \mathcal{C}[\tau_1] : \mathtt{nat}$ и $\emptyset \vdash \mathcal{C}[\tau_2] : \mathtt{nat}$, то

$$\mathcal{C}[\tau_1] \leq_{op} \mathcal{C}[\tau_2] : \mathtt{nat}.$$

Ще пишем $\tau_1 \cong_{ctx} \tau_2 : \mathtt{a}$, ако $\tau_1 \leq_{ctx} \tau_2 : \mathtt{a}$ и $\tau_2 \leq_{ctx} \tau_1 : \mathtt{a}$.

Възможно е по-общо да се дефинира релацията $\leq_{ctx}$ не само за затворени, но и за произволни добре типизирани термове. За нашите цели е достатъчно да се ограничим само до затворени термове.

Two phrases of a programming language are contextually equivalent if any occurrences of the first phrase in a complete program can be replaced by the second phrase without affecting the observable results of executing the program. This kind of program equivalence is also known as operational, or observational equivalence [6, стр. 44].

Денотационната семантика $[\![.]\!]$ се нарича **напълно абстрактна**, ако денотационната и операционната наредба съвпадат, т.е. за всеки два затворени терма τ_1 и τ_2 от тип $\mathtt{a}$ е изпълнено, че:

$$\tau_1 \leq_{den} \tau_2 : \mathtt{a} \text{ точно тогава, когато } \tau_1 \leq_{ctx} \tau_2 : \mathtt{a}.$$

Един от основните ранни резултати в изучаването на семантиката на езици за програмиране е, че нашата денотационна семантика не е напълно абстрактна. Практически, ако имаме два терма, които са контекстно еквивалентни, то можем да заменим единия с другия в произволна програма, без да има видими разлики на ниво изпълнение на програмата. Проблемът е, че с този формализъм е трудно да се работи, защото в дефиницията имаме квантор за всеобщност относно всички контексти (програмни фрагменти).

Твърдение 4.10. За произволни затворени термове τ_1 и τ_2 от един и същ тип е изпълнено, че:

(1) $\tau_1 \leq_{ctx} \tau_2 : \mathtt{nat} \implies \tau_1 \leq_{op} \tau_2 : \mathtt{nat}$.

(2) $\tau_1 \leq_{ctx} \tau_2 : \mathtt{b} \to \mathtt{c} \implies (\forall \rho \in \mathrm{PCF}_{\mathtt{b}})[\tau_1\rho \leq_{ctx} \tau_2\rho : \mathtt{c}]$.

Упътване.

(1) Просто вземете контекст $\mathcal{C} \stackrel{\text{деф}}{=} -$. Ясно е, че $\mathcal{C}[\tau_i] \equiv \tau_i$.

(2) Да фиксираме произволен терм $\rho \in \mathrm{PCF}_{\mathtt{b}}$. Да разгледаме произволен контекст $\mathcal{C}[-]$, за който $\mathcal{C}[\tau_1\rho]$ и $\mathcal{C}[\tau_2\rho]$ са затворени термове от тип $\mathtt{nat}$. Разглеждаме контекста $\mathcal{C}' \stackrel{\text{деф}}{=} \mathcal{C}[-\rho]$, т.е. заменяме $-$ с $-\rho$ в контекста $\mathcal{C}$. Лесно се съобразява, че

$$\mathcal{C}'[\tau_i] \equiv \mathcal{C}[\tau_i\rho].$$

Понеже $\tau_1 \leq_{ctx} \tau_2 : \mathtt{b} \to \mathtt{c}$, то за специалния контекст $\mathcal{C}'$ имаме, че

$$(\forall \mathtt{n})[\, \mathcal{C}'[\tau_1] \Downarrow_{\mathtt{nat}} \mathtt{n} \implies \mathcal{C}'[\tau_2] \Downarrow_{\mathtt{nat}} \mathtt{n}\,].$$

Оттук веднага следва, че за първоначалния контекст $\mathcal{C}$ имаме, че

$$(\forall \mathtt{n})[\, \mathcal{C}[\tau_1\rho] \Downarrow_{\mathtt{nat}} \mathtt{n} \implies \mathcal{C}[\tau_2\rho] \Downarrow_{\mathtt{nat}} \mathtt{n}\,].$$

□

Твърдение 4.11. За произволни затворени термове τ_1 и τ_2 от един и същ тип $\mathtt{a}$ е изпълнено, че:

$$\tau_1 \leq_{ctx} \tau_2 : \mathtt{a} \iff (\forall \rho \in \mathrm{PCF}_{\mathtt{a}\to\mathtt{nat}})[\, \rho\tau_1 \leq_{op} \rho\tau_2 : \mathtt{nat}\,].$$

Доказателство. За ($\Rightarrow$), при даден терм $\rho \in \mathrm{PCF}_{\mathtt{a}\to\mathtt{nat}}$, просто вземете контекстът $\mathcal{C}$ да бъде $\mathcal{C} \stackrel{\text{деф}}{=} \rho\ -$. Тогава е ясно, че $\mathcal{C}[\tau_1]$ и $\mathcal{C}[\tau_2]$ са завършени програми и следователно, щом $\tau_1 \leq_{ctx} \tau_2 : \mathtt{a}$, то е изпълнено, че:

$$(\forall \mathtt{n})[\, \underbrace{\mathcal{C}[\tau_1]}_{\rho\tau_1} \Downarrow_{\mathtt{nat}} \mathtt{n} \implies \underbrace{\mathcal{C}[\tau_2]}_{\rho\tau_2} \Downarrow_{\mathtt{nat}} \mathtt{n}\,].$$

За ($\Leftarrow$), нека разгледаме произволен контекст $\mathcal{C}[-]$, за който $\mathcal{C}[\tau_1] \Downarrow_{\mathtt{nat}} \mathtt{n}$. Трябва да докажем, че $\mathcal{C}[\tau_2] \Downarrow_{\mathtt{nat}} \mathtt{n}$.

Нека $\mathtt{z}$ е променлива, която не се среща в $\mathcal{C}[-]$. Да разгледаме терма $\rho' \stackrel{\text{деф}}{=} \mathcal{C}[-/\mathtt{z}]$, т.е. в контекста $\mathcal{C}$ заменяме $-$ с нова променлива $\mathtt{z}$. Ясно е, че $\mathtt{z} : \mathtt{a} \vdash \rho' : \mathtt{nat}$. Разгледайте затворения терм $\rho \stackrel{\text{деф}}{=} \lambda\, \mathtt{z} : \mathtt{a}\, .\, \rho'$ от тип $\mathtt{a} \to \mathtt{nat}$. Тогава, понеже τ_i са затворени термове, то $\mathcal{C}[\tau_i] \equiv \rho'[\mathtt{z}/\tau_i]$.

Да напомним, че от правилата на операционната семантика имаме, че

$$\dfrac{(\text{val})\ \dfrac{\rho \text{ е стойност}}{\rho \Downarrow_{\mathtt{a}\to\mathtt{nat}} \rho} \qquad \rho'[\mathtt{z}/\tau_1] \Downarrow_{\mathtt{nat}} \mathtt{n}}{\rho\tau_1 \Downarrow_{\mathtt{nat}} \mathtt{n}}\ (\text{cbn})$$

Това означава, че имаме и $\rho\tau_2 \Downarrow_{\mathtt{nat}} \mathtt{n}$. Тогава пак според правилата на операционната семантика е ясно, че със сигурност имаме и $\rho'[\mathtt{z}/\tau_2] \Downarrow_{\mathtt{nat}} \mathtt{n}$. Така доказахме, че $\tau_1 \leq_{ctx} \tau_2 : \mathtt{a}$. □

Сега започва да става интересно, защото ще свържем логическата релация $\lhd_{\mathtt{a}}$ с релацията $\leq_{ctx}$.

Твърдение 4.12. Нека τ_1 и τ_2 са произволни затворени термове от тип $\mathtt{a}$ и d е произволен елемент на областта на Скот $[\![\mathtt{a}]\!]$. Тогава е изпълнена импликацията:

$$(d \lhd_{\mathtt{a}} \tau_1 \ \&\ \tau_1 \leq_{ctx} \tau_2 : \mathtt{a}) \implies d \lhd_{\mathtt{a}} \tau_2.$$

Доказателство. Доказателството следва дефиницията на логическата релация $\lhd_{\mathtt{a}}$, което означава, че ще направим индукция по построението на типовете $\mathtt{a}$.

Нека $\mathtt{a} = \mathtt{nat}$. Да приемем, че $d \neq \bot$, защото е ясно, че $\bot \lhd_{\mathtt{nat}} \tau_2$. Понеже $d \lhd_{\mathtt{nat}} \tau_1$, то от дефиницията на $\lhd_{\mathtt{nat}}$ веднага следва, че $\tau_1 \Downarrow_{\mathtt{nat}} \mathtt{d}$. Понеже $\tau_1 \leq_{ctx} \tau_2 : \mathtt{nat}$, то според (1) на *Твърдение* 4.10 имаме, че $\tau_2 \Downarrow_{\mathtt{nat}} \mathtt{d}$. Заключаваме, че $d \lhd_{\mathtt{nat}} \tau_2$.

Нека $\mathtt{a} = \mathtt{b} \to \mathtt{c}$ и да приемем, че имаме $d \lhd_{\mathtt{b}\to\mathtt{c}} \tau_1$ и $\tau_1 \leq_{ctx} \tau_2 : \mathtt{b} \to \mathtt{c}$. Трябва да докажем, че $d \lhd_{\mathtt{b}\to\mathtt{c}} \tau_2$, което според дефиницията на логическата релация $\lhd_{\mathtt{b}\to\mathtt{c}}$ означава, че за произволни $u \in [\![\mathtt{b}]\!]$ и $\rho \in \mathrm{PCF}_{\mathtt{b}}$, за които $u \lhd_{\mathtt{b}} \rho$, трябва да докажем, че $d(u) \lhd_{\mathtt{c}} \tau_2\rho$. Да разгледаме произволен такъв елемент u и терм ρ. От $d \lhd_{\mathtt{b}\to\mathtt{c}} \tau_1$ имаме, че $d(u) \lhd_{\mathtt{c}} \tau_1\rho$. От (2) на *Твърдение* 4.10 имаме, че щом $\tau_1 \leq_{ctx} \tau_2 : \mathtt{b} \to \mathtt{c}$, то $\tau_1\rho \leq_{ctx} \tau_2\rho : \mathtt{c}$. Сега от **(И.П.)** заключаваме, че $d(u) \lhd_{\mathtt{c}} \tau_2\rho$. □

Твърдение 4.13. За произволни затворени термове τ_1 и τ_2 от тип $\mathtt{a}$,

$$\tau_1 \leq_{ctx} \tau_2 : \mathtt{a} \iff [\![\tau_1]\!] \lhd_{\mathtt{a}} \tau_2.$$

Доказателство. $(\Rightarrow)$ Нека $\tau_1 \leq_{ctx} \tau_2 : \mathtt{a}$. От *Твърдение* 4.9 имаме, че

$$[\![\tau_1]\!] \lhd_{\mathtt{a}} \tau_1.$$

Тогава от *Твърдение* 4.12 директно следва, че $[\![\tau_1]\!] \lhd_{\mathtt{a}} \tau_2$.

$(\Leftarrow)$ Нека сега $[\![\tau_1]\!] \lhd_{\mathtt{a}} \tau_2$. Тук на помощ ни идва характеризацията на $\leq_{ctx}$ от *Твърдение* 4.11. Да разгледаме произволен терм $\rho \in \mathrm{PCF}_{\mathtt{a}\to\mathtt{nat}}$. Трябва да докажем, че:

$$(\forall \mathtt{n})[\,\rho\tau_1 \Downarrow_{\mathtt{nat}} \mathtt{n} \implies \rho\tau_2 \Downarrow_{\mathtt{nat}} \mathtt{n}\,].$$

Според *Твърдение* 4.9 имаме, че $[\![\rho]\!] \lhd_{\mathtt{a}\to\mathtt{nat}} \rho$. Тогава от дефиницията на релацията $\lhd_{\mathtt{a}\to\mathtt{nat}}$ следва, че за произволен елемент e от областта на Скот $[\![\mathtt{a}]\!]$ и произволен затворен терм $\mu : \mathtt{a}$, ако $e \lhd_{\mathtt{a}} \mu$, то $[\![\rho]\!](e) \lhd_{\mathtt{nat}} \rho\mu$. Нека да положим $e \stackrel{\text{деф}}{=} [\![\tau_1]\!]$ и $\mu \stackrel{\text{деф}}{=} \tau_2$. Тогава получаваме, че:

$$\underbrace{[\![\rho\tau_1]\!]}_{[\![\rho]\!]([\![\tau_1]\!])} \lhd_{\mathtt{nat}} \rho\tau_2.$$

Сега за произволна стойност $\mathtt{n}$ имаме, че:

$$\begin{aligned} \rho\tau_1 \Downarrow_{\mathtt{nat}} \mathtt{n} &\implies [\![\rho\tau_1]\!] = n && \text{// Теорема за коректност} \\ &\implies \rho\tau_2 \Downarrow_{\mathtt{nat}} \mathtt{n}. && \text{// от } [\![\rho\tau_1]\!] \lhd_{\mathtt{nat}} \rho\tau_2 \end{aligned}$$

Заключаваме, че $\tau_1 \leq_{ctx} \tau_2 : \mathtt{a}$. □

Твърдение 4.14. За произволни затворени термове τ_1 и τ_2 е изпълнено, че:

(1) $\tau_1 \leq_{ctx} \tau_2 : \mathtt{nat} \iff \tau_1 \leq_{op} \tau_2 : \mathtt{nat}$;

(2) $\tau_1 \leq_{ctx} \tau_2 : \mathtt{a} \to \mathtt{b} \iff (\forall \rho \in \mathrm{PCF}_{\mathtt{a}})[\, \tau_1\rho \leq_{ctx} \tau_2\rho : \mathtt{b}\,]$.

Доказателство.

(1) Посоката $(\Rightarrow)$ следва директно от (1) на *Твърдение* 4.10. За посоката $(\Leftarrow)$, понеже

$$\begin{aligned} [\![\tau_1]\!] = [\![\mathtt{n}]\!] &\implies \tau_1 \Downarrow_{\mathtt{nat}} \mathtt{n} && \text{// Теорема за адекватност} \\ &\implies \tau_2 \Downarrow_{\mathtt{nat}} \mathtt{n}, && \text{// от условието} \end{aligned}$$

то получаваме, че $[\![\tau_1]\!] \lhd_{\mathtt{nat}} \tau_2$. Тогава от *Твърдение* 4.13 получаваме, че $\tau_1 \leq_{ctx} \tau_2 : \mathtt{nat}$.

(2) Посоката $(\Rightarrow)$ е на практика (2) на *Твърдение* 4.10. За посоката $(\Leftarrow)$, според *Твърдение* 4.13, достатъчно е да докажем, че $[\![\tau_1]\!] \lhd_{\mathtt{a}\to\mathtt{b}} \tau_2$. Според дефиницията на логическата релация $\lhd_{\mathtt{a}\to\mathtt{b}}$, това означава да докажем че за произволни $u \in [\![\mathtt{a}]\!]$ и $\rho \in \mathrm{PCF}_{\mathtt{a}}$, за които $u \lhd_{\mathtt{a}} \rho$, то е изпълнено, че $[\![\tau_1]\!](u) \lhd_{\mathtt{b}} \tau_2\rho$. Да разгледаме произволни такъв елемент u и терм ρ. Тъй като от *Следствие* 4.3 знаем, че $[\![\tau_1]\!] \lhd_{\mathtt{a}\to\mathtt{b}} \tau_1$, то $[\![\tau_1]\!](u) \lhd_{\mathtt{b}} \tau_1\rho$. Но понеже $\tau_1\rho \leq_{ctx} \tau_2\rho : \mathtt{b}$, то от *Твърдение* 4.12 следва, че $[\![\tau_1]\!](u) \lhd_{\mathtt{b}} \tau_2\rho$.

□

Задача 4.4. Докажете, че винаги можем да направим следния извод:

$$\frac{\tau_1 \leq_{ctx} \tau_2 : \mathtt{a} \to \mathtt{b} \qquad \rho_1 \leq_{ctx} \rho_2 : \mathtt{a}}{\tau_1\rho_1 \leq_{ctx} \tau_2\rho_2 : \mathtt{b}}$$

Упътване. Според *Твърдение* 4.13, от $\tau_1 \leq_{ctx} \tau_2 : \mathtt{a} \to \mathtt{b}$ имаме, че $[\![\tau_1]\!] \lhd_{\mathtt{a}\to\mathtt{b}} \tau_2$. Тогава, според дефиницията на $\lhd_{\mathtt{a}\to\mathtt{b}}$, за произволен елемент $u \in [\![\mathtt{a}]\!]$ и произволен $\mu \in \mathrm{PCF}_{\mathtt{a}}$, за които $u \lhd_{\mathtt{a}} \mu$ е изпълнено, че $[\![\tau_1]\!](u) \lhd_{\mathtt{b}} \tau_2\mu$.

От $\rho_1 \leq_{ctx} \rho_2 : \mathtt{a}$ имаме, че $[\![\rho_1]\!] \lhd_{\mathtt{b}} \rho_2$. Нека положим $u \stackrel{\text{деф}}{=} [\![\rho_1]\!]$ и $\mu \stackrel{\text{деф}}{=} \rho_2$. Тогава получаваме, че $[\![\tau_1]\!]([\![\rho_1]\!]) \lhd_{\mathtt{b}} \tau_2\rho_2$. Сега, понеже $[\![\tau_1]\!]([\![\rho_1]\!]) = [\![\tau_1\rho_1]\!]$, отново от *Твърдение* 4.13, заключаваме, че $\tau_1\rho_1 \leq_{ctx} \tau_2\rho_2 : \mathtt{b}$. □

Естествено е да се запитаме дали можем да разширим (1) на *Твърдение* 4.14 за по-сложни от $\mathtt{nat}$ типове $\mathtt{a}$, т.е. възможно ли е

$$\tau_1 \leq_{ctx} \tau_2 : \mathtt{a} \iff \tau_1 \leq_{op} \tau_2 : \mathtt{a}?$$

Би било странно, ако можем, защото това би обезсмилило разглеждането на релацията $\leq_{ctx}$. Първо в *Твърдение* 4.15 ще видим, че винаги имаме импликацията ($\Leftarrow$), но по-късно в *Твърдение* 4.17 ще видим, че дори за типа $\mathtt{a} = \mathtt{nat} \to \mathtt{nat}$ нямаме импликация ($\Rightarrow$).

Твърдение 4.15. Докажете, че за всеки два затворени терма τ_1 и τ_2 е изпълнена импликацията:

$$\tau_1 \leq_{op} \tau_2 : \mathtt{a} \implies \tau_1 \leq_{ctx} \tau_2 : \mathtt{a}.$$

Доказателство. Имаме всичко необходимо за да докажем това твърдение:

$$\dfrac{\dfrac{\dfrac{\text{от условието}}{\tau_1 \leq_{op} \tau_2 : \mathtt{a}} \quad \dfrac{\textit{Следствие } 4.3}{[\![\tau_1]\!] \lhd_{\mathtt{a}} \tau_1}}{[\![\tau_1]\!] \lhd_{\mathtt{a}} \tau_2}\ (\textit{Твърдение } 4.9)}{\tau_1 \leq_{ctx} \tau_2 : \mathtt{a}}\ (\textit{Твърдение } 4.13)$$

□

Твърдение 4.16. За произволни затворени термове τ_1 и τ_2 е изпълнена импликацията:

$$\tau_1 \leq_{den} \tau_2 : \mathtt{a} \implies \tau_1 \leq_{ctx} \tau_2 : \mathtt{a}.$$

Доказателство. Според *Твърдение* 4.11, достатъчно е да докажем, че за произволен терм $\rho \in \mathrm{PCF}_{\mathtt{a}\to\mathtt{nat}}$ имаме импликацията:

$$(\forall \mathtt{n})[\, \rho\tau_1 \Downarrow_{\mathtt{nat}} \mathtt{n} \implies \rho\tau_2 \Downarrow_{\mathtt{nat}} \mathtt{n}\,].$$

Но понеже $[\![\tau_1]\!] \sqsubseteq [\![\tau_2]\!]$, това е лесно:

$$\begin{aligned}
\rho\tau_1 \Downarrow_{\mathtt{nat}} \mathtt{n} &\implies [\![\rho\tau_1]\!] = [\![\mathtt{n}]\!] && \text{// Теорема за коректност}\\
&\implies [\![\rho]\!]([\![\tau_1]\!]) = [\![\mathtt{n}]\!]\\
&\implies [\![\rho]\!]([\![\tau_2]\!]) = [\![\mathtt{n}]\!] && \text{// монотонност на } [\![\rho]\!]\\
&\implies [\![\rho\tau_2]\!] = [\![\mathtt{n}]\!]\\
&\implies \rho\tau_2 \Downarrow_{\mathtt{nat}} \mathtt{n}. && \text{// Теорема за адекватност}
\end{aligned}$$

□

Можем да обобщим всичко, което сме направили до момента като формулираме следната теорема.

Теорема 4.4. За всички затворени термове τ_1 и τ_2 е изпълнено, че:

(1) $\tau_1 \cong_{op} \tau_2 : \mathtt{a} \implies \tau_1 \cong_{ctx} \tau_2 : \mathtt{a}$;

(2) $\tau_1 \cong_{den} \tau_2 : \mathtt{a} \implies \tau_1 \cong_{ctx} \tau_2 : \mathtt{a}$.

В частния случай на $\mathtt{a} = \mathtt{nat}$ имаме и обратните импликации на *Теорема* 4.4.

Следствие 4.6. За всички затворени термове τ_1 и τ_2 от тип $\mathtt{nat}$ са изпълнени еквивалентностите:

(1) $\tau_1 \cong_{op} \tau_2 : \mathtt{nat} \iff \tau_1 \cong_{ctx} \tau_2 : \mathtt{nat}$;

(2) $\tau_1 \cong_{den} \tau_2 : \mathtt{nat} \iff \tau_1 \cong_{ctx} \tau_2 : \mathtt{nat}$.

Доказателство. Свойство (1) представлява точно (1) на *Твърдение* 4.14. За обратната посока на (2), нека $\tau_1 \leq_{ctx} \tau_2 : \mathtt{nat}$. Трябва да докажем, че $[\![\tau_1]\!] \sqsubseteq [\![\tau_2]\!]$. Нека $[\![\tau_1]\!] = n \neq \bot$, защото ако $[\![\tau_1]\!] = \bot$, то е ясно, че $[\![\tau_1]\!] \sqsubseteq [\![\tau_2]\!]$. Имаме импликациите:

$$\begin{aligned}
[\![\tau_1]\!] = n &\implies \tau_1 \Downarrow_{\mathtt{nat}} \mathtt{n} && \text{// Теорема за адекватност}\\
&\implies \tau_2 \Downarrow_{\mathtt{nat}} \mathtt{n} && \text{// (1) на Твърдение 4.14}\\
&\implies [\![\tau_2]\!] = n. && \text{// Теорема за коректност}
\end{aligned}$$

□

Нека сега да видим, че нямаме обратните импликации на *Следствие* 4.6 за по-високи типове от $\mathtt{nat}$ като разгледаме термовете $\Omega_{\mathtt{a}}$ и $\Omega'_{\mathtt{a}}$, които сме срещали и преди и сме разглеждали техните свойства в *Задача* 4.2.

Твърдение 4.17. $\Omega'_{\mathtt{nat}} \cong_{ctx} \Omega_{\mathtt{nat}\to\mathtt{nat}} : \mathtt{nat} \to \mathtt{nat}$.

Доказателство.
Понеже ние вече знаем, че $[\![\Omega'_{\mathtt{nat}}]\!] = [\![\Omega_{\mathtt{nat}\to\mathtt{nat}}]\!]$, то според (2) на *Теорема* 4.4 следва, че $\Omega'_{\mathtt{nat}} \cong_{ctx} \Omega_{\mathtt{nat}\to\mathtt{nat}}$. □

Да напомним, че

$$\Omega_{\mathtt{a}} \overset{\text{деф}}{=} \mathtt{fix}(\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mathtt{x})$$
$$\Omega'_{\mathtt{a}} \overset{\text{деф}}{=} \lambda\, \mathtt{x} : \mathtt{a} \,.\, \Omega_{\mathtt{a}}.$$

Знаем, че $\Omega_{\mathtt{nat}\to\mathtt{nat}} \not\Downarrow_{\mathtt{nat}\to\mathtt{nat}}$, но $\Omega'_{\mathtt{nat}} \Downarrow_{\mathtt{nat}\to\mathtt{nat}} \Omega'_{\mathtt{nat}}$. Това означава, че според *Твърдение* 4.17, термовете $\Omega_{\mathtt{nat}\to\mathtt{nat}}$ и $\Omega'_{\mathtt{nat}}$ ни дават пример кога нямаме обратната импликация в (1) на *Теорема* 4.4. Остава ни да продължим да търсим термове τ_1 и τ_2 от някакъв тип a, за които $[\![\tau_1]\!] \neq [\![\tau_2]\!]$ и $\tau_1 \cong_{ctx} \tau_2 : \mathtt{a}$.

Оттук е очевидно, че също така имаме и следното важно наблюдение.

Забележка 4.2. За произволен тип a е изпълнено, че:

$$\Omega'_{\mathtt{a}} \not\leq_{op} \Omega_{\mathtt{a}\to\mathtt{a}} : \mathtt{a} \to \mathtt{a},$$

но

$$\Omega'_{\mathtt{a}} \leq_{den} \Omega_{\mathtt{a}\to\mathtt{a}} : \mathtt{a} \to \mathtt{a}$$

и

$$\Omega'_{\mathtt{a}} \leq_{ctx} \Omega_{\mathtt{a}\to\mathtt{a}} : \mathtt{a} \to \mathtt{a}.$$

Интересно е да отбележим, че за един прост вид типове ние имаме обратната импликация в (2) на *Теорема* 4.4.

Тази задача е давана на изпит в Кеймбридж 2010 г. за случая $\mathtt{a} = \mathtt{nat} \to \mathtt{nat}$.

Задача 4.5. Нека $\mathtt{a} = \mathtt{nat} \to \mathtt{nat} \to \cdots \to \mathtt{nat}$. Докажете, че

$$\tau_1 \cong_{ctx} \tau_2 : \mathtt{a} \implies \tau_1 \cong_{den} \tau_2 : \mathtt{a}.$$

Доказателство. За да проверим това свойство е достатъчно да проверим, че ако $\tau_1 \leq_{ctx} \tau_2 : \mathtt{a}$, то $\tau_1 \leq_{den} \tau_2 : \mathtt{a}$.

Ще проведем доказателството с индукция по построението на типа a. Нека $\mathtt{a} = \mathtt{nat}$. Тогава импликацията следва директно от (2) на *Следствие* 4.6.

Нека сега да представим типа a като $\mathtt{nat} \to \mathtt{b}$. По условие имаме, че $\tau_1 \leq_{ctx} \tau_2 : \mathtt{nat} \to \mathtt{b}$. Според (2) на *Твърдение* 4.14, това означава, че имаме за всеки затворен терм ρ от тип nat, $\tau_1\rho \leq_{ctx} \tau_2\rho : \mathtt{b}$. От **(И.П.)** за типа b следва, че $\tau_1 \leq_{den} \tau_2 : \mathtt{b}$ или с други думи,

Да напомним, че

$$\mathtt{a} = \mathtt{nat} \to \mathtt{nat} \to \mathtt{nat}$$

е всъщност

$$\mathtt{a} = \mathtt{nat} \to (\mathtt{nat} \to \mathtt{nat}).$$

Поради тази причина, в индукционното предположение, не можем да разглеждаме a като $\mathtt{b} \to \mathtt{nat}$.

$$[\![\tau_1\rho]\!] \sqsubseteq [\![\tau_2\rho]\!]. \tag{4.4}$$

Нека разгледаме произволен елемент $n \in \mathbb{N}_\perp$.

Тук важното е това, че всеки елемент на $\mathbb{N}_\perp$ е определим с терм в езика PCF.

- Ако $n \neq \perp$, то нека положим $\rho \stackrel{\text{деф}}{=} \mathtt{n}$.
- Ако $n = \perp$, то нека положим $\rho \stackrel{\text{деф}}{=} \Omega_{\mathtt{nat}}$.

Ясно е, че сме избрали затворения терм ρ от тип nat така, че $[\![\rho]\!] = n$. Достатъчно е да проследим следните импликации

$$\begin{aligned}[\![\tau_1]\!](n) &= [\![\tau_1]\!]([\![\rho]\!])\\ &= [\![\tau_1\rho]\!]\end{aligned}$$

$$\begin{aligned} &\sqsubseteq [\![\tau_2\rho]\!] && \text{// от (4.4)} \\ &= [\![\tau_2]\!]([\![\rho]\!]) \\ &= [\![\tau_2]\!](n). \end{aligned}$$

□

Това означава, че ако искаме да намерим затворени термове τ_1 и τ_2 от тип $\mathtt{a}$, за които $\tau_1 \not\cong_{den} \tau_2 : \mathtt{a}$, но $\tau_1 \cong_{ctx} \tau_2 : \mathtt{a}$, то трябва да разгледаме по-сложни типове $\mathtt{a}$.

Задача 4.6. Докажете или опровергайте твърденията:

(1) $\lambda\, \mathtt{x} : \mathtt{nat} \,.\, \mathtt{x} \cong_{ctx} \lambda\, \mathtt{x} : \mathtt{nat} \,.\, \mathtt{x + 0} : \mathtt{nat} \to \mathtt{nat}$;

(2) $\lambda\, \mathtt{x} : \mathtt{nat} \,.\, \mathtt{x} \cong_{ctx} \lambda\, \mathtt{x} : \mathtt{nat} \,.\, \mathtt{(x - 1) + 1} : \mathtt{nat} \to \mathtt{nat}$;

Задача в Кеймбридж 2020 г. [5]

Задача 4.7. Докажете или опровергайте твърдението, че за всеки два типа $\mathtt{a}$ и $\mathtt{b}$ и всеки два затворени терма τ_1 от тип $\mathtt{a} \to \mathtt{b}$ и τ_2 от тип $\mathtt{b} \to \mathtt{a}$ е изпълнено следното:

$$\mathtt{fix}(\lambda\, \mathtt{y} : \mathtt{b} \,.\, \tau_1(\tau_2(\mathtt{y}))) \cong_{ctx} \tau_1(\mathtt{fix}(\lambda\, \mathtt{x} : \mathtt{a} \,.\, \tau_2(\tau_1(\mathtt{x})))) : \mathtt{b}.$$

Упътване. Използвайте *Задача* 1.21. □

Задача 4.8. Нека τ_1 и τ_2 са затворени термове от тип $\mathtt{a} \to \mathtt{b}$ и нека μ_1 и μ_2 са затворени термове от тип $\mathtt{b} \to \mathtt{c}$, за които:

$$\begin{aligned} &\tau_1 \cong_{ctx} \tau_2 : \mathtt{a} \to \mathtt{b} \\ &\mu_1 \cong_{ctx} \mu_2 : \mathtt{b} \to \mathtt{c}. \end{aligned}$$

Докажете или опровергайте, че тогава имаме еквивалентността:

$$\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mu_1(\tau_1\mathtt{x}) \cong_{ctx} \lambda\, \mathtt{x} : \mathtt{a} \,.\, \mu_2(\tau_2\mathtt{x}) : \mathtt{a} \to \mathtt{c}. \tag{4.5}$$

Задача в Кеймбридж 2015 г. [5]

Упътване. За да докажем, че еквивалентността (4.5) е изпълнена, според (2) на *Твърдение* 4.14 е достатъчно да докажем, че за произволен затворен терм ρ от тип $\mathtt{a}$ е изпълнено, че:

$$(\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mu_1(\tau_1\mathtt{x}))\rho \cong_{ctx} (\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mu_2(\tau_2\mathtt{x}))\rho : \mathtt{c}.$$

Сега, ще използваме равенствата за $i = 1, 2$:

$$[\![(\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mu_i(\tau_i\mathtt{x}))\rho]\!] = [\![\mu_i]\!]([\![\tau_i]\!]([\![\rho]\!])) = [\![\mu_i(\tau_i\rho)]\!],$$

откъдето според (2) на *Теорема* 4.4 следва, че

$$(\lambda\, \mathtt{x} : \mathtt{a} \,.\, \mu_i(\tau_i\mathtt{x}))\rho \cong_{ctx} \mu_i(\tau_i\rho).$$

Оттук следва, че е достатъчно да докажем, че

$$\mu_1(\tau_1\rho) \cong_{ctx} \mu_2(\tau_2\rho) : \mathtt{c}. \tag{4.6}$$

Но това е лесно да се види, защото, щом $\tau_1 \cong_{ctx} \tau_2 : \mathtt{a} \to \mathtt{b}$, то според (2) на *Твърдение* 4.14 имаме, че $\tau_1\rho \cong_{ctx} \tau_2\rho : \mathtt{b}$, Сега прилагаме *Задача* 4.4 и оттам веднага получаваме (4.6). □

4.9 Езикът PCF(bool)

Всъщност в [7, Глава 4.1] това е „истинската” дефиниция на езика PCF.

- Типове
$$\mathtt{a} ::= \mathtt{bool} \mid \mathtt{nat} \mid \mathtt{a} \to \mathtt{a}$$

- Изрази
$$\begin{aligned}\tau ::= {} & \mathtt{true} \mid \mathtt{false} \mid \mathtt{n} \mid \mathtt{x} \mid \tau_1 + \tau_2 \mid \tau_1 - \tau_2 \mid \tau_1 \mathtt{==} \tau_2 \mid \\ & \mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3 \mid \tau_1\tau_2 \mid \lambda\, \mathtt{x} : \mathtt{a} \,.\, \tau_1 \mid \mathtt{fix}(\tau_1).\end{aligned}$$

- Стойностите са затворени термове от следния вид:
$$\mathtt{v} ::= \mathtt{true} \mid \mathtt{false} \mid \mathtt{n} \mid \lambda\, \mathtt{x} : \mathtt{a} \,.\, \mu$$

4.9.1 Типизираща релация

Релацията $\Gamma \vdash \tau : \mathtt{a}$ за езика PCF(bool)е почти същата както за езика PCF. Имаме две нови правила:

$$\frac{}{\Gamma \vdash \mathtt{true} : \mathtt{bool}}\ (\text{true}) \qquad\qquad \frac{}{\Gamma \vdash \mathtt{false} : \mathtt{bool}}\ (\text{false})$$

Имаме и две променени правила:

$$\frac{\Gamma \vdash \tau_1 : \mathtt{nat} \qquad \Gamma \vdash \tau_2 : \mathtt{nat}}{\Gamma \vdash \tau_1 \mathtt{==} \tau_2 : \mathtt{bool}}\ (\text{eq})$$

$$\frac{\Gamma \vdash \tau_1 : \mathtt{bool} \qquad \Gamma \vdash \tau_2 : \mathtt{a} \qquad \Gamma \vdash \tau_3 : \mathtt{a}}{\Gamma \vdash \mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3 : \mathtt{a}}\ (\text{if})$$

Всички останали правила са същите.

4.9.2 Операционна семантика

Тук отново всичко е почти същото както преди със следните разлики

$$\frac{\tau_1 \Downarrow^{\ell_1}_{\mathtt{nat}} \mathtt{v}_1 \qquad \tau_3 \Downarrow^{\ell_2}_{\mathtt{a}} \mathtt{v}_2 \qquad \mathtt{v}_1 \equiv \mathtt{v}_2}{\tau_1 \mathtt{==} \tau_2 \Downarrow^{\ell_1+\ell_2+1}_{\mathtt{bool}} \mathtt{true}} \qquad \frac{\tau_1 \Downarrow^{\ell_1}_{\mathtt{nat}} \mathtt{v}_1 \qquad \tau_3 \Downarrow^{\ell_2}_{\mathtt{a}} \mathtt{v}_2 \qquad \mathtt{v}_1 \not\equiv \mathtt{v}_2}{\tau_1 \mathtt{==} \tau_2 \Downarrow^{\ell_1+\ell_2+1}_{\mathtt{bool}} \mathtt{false}}$$

$$\frac{\tau_1 \Downarrow^{\ell_1}_{\mathtt{bool}} \mathtt{false} \qquad \tau_3 \Downarrow^{\ell_2}_{\mathtt{a}} \mathtt{v}}{\mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3 \Downarrow^{\ell_1+\ell_2+1}_{\mathtt{a}} \mathtt{v}} \qquad \frac{\tau_1 \Downarrow^{\ell_1}_{\mathtt{bool}} \mathtt{true} \qquad \tau_2 \Downarrow^{\ell_2}_{\mathtt{a}} \mathtt{v}}{\mathtt{if}\ \tau_1\ \mathtt{then}\ \tau_2\ \mathtt{else}\ \tau_3 \Downarrow^{\ell_1+\ell_2+1}_{\mathtt{a}} \mathtt{v}}$$

Фигура 4.10: Променени в правилата на операционната семантика за PCF(bool)

4.9.3 Денотационна семантика

В [7, Глава 4.3] се нарича *standard fixed-point semantics* of PCF.

Семантиката на всеки тип ще бъде област на Скот както следва:

$$[\![\texttt{bool}]\!] \stackrel{\text{деф}}{=} \mathbb{B}_\perp = \{true, false\}_\perp.$$

- Нека $\tau \equiv \texttt{true}$. Тогава
$$[\![\texttt{true}]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} true.$$

- Нека $\tau \equiv \texttt{false}$. Тогава
$$[\![\texttt{false}]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} false.$$

- Нека $\tau \equiv \tau_1 \texttt{ == } \tau_2$. Тогава
$$[\![\tau_1 \texttt{ == } \tau_2]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} \begin{cases} true, & \text{ако } [\![\tau_1]\!]_\Gamma(\overline{u}) = [\![\tau_1]\!]_\Gamma(\overline{u}) \in \mathbb{N} \\ false, & \text{ако } [\![\tau_1]\!]_\Gamma(\overline{u}) \neq [\![\tau_1]\!]_\Gamma(\overline{u}) \in \mathbb{N} \\ \perp, & \text{ако } [\![\tau_1]\!]_\Gamma(\overline{u}) = \perp \text{ или } [\![\tau_1]\!]_\Gamma(\overline{u}) = \perp. \end{cases}$$

- Нека $\tau \equiv \texttt{if } \tau_1 \texttt{ then } \tau_2 \texttt{ else } \tau_3$. Тогава
$$[\![\texttt{if } \tau_1 \texttt{ then } \tau_2 \texttt{ else } \tau_3]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} \begin{cases} [\![\tau_2]\!]_\Gamma(\overline{u}), & \text{ако } [\![\tau_1]\!]_\Gamma(\overline{u}) = true \\ [\![\tau_3]\!]_\Gamma(\overline{u}), & \text{ако } [\![\tau_1]\!]_\Gamma(\overline{u}) = false \\ \perp, & \text{ако } [\![\tau_1]\!]_\Gamma(\overline{u}) = \perp. \end{cases}$$

Теорема 4.5 (Теорема за коректност за езика PCF(bool))**.** Докажете, че за всеки затворен терм τ от тип $\texttt{a}$ и стойност $\texttt{v}$, е изпълнена импликацията:

$$\tau \Downarrow_{\texttt{a}} \texttt{v} \implies [\![\tau]\!] = [\![\texttt{v}]\!] \in [\![\texttt{b}]\!].$$

Теорема 4.6 (Теорема за адекватност за езика PCF(bool))**.** Нека разгледаме тип $\texttt{a} = \texttt{nat}$ или $\texttt{a} = \texttt{bool}$. За всеки затворен терм τ от тип $\texttt{a}$ е изпълнена импликацията

$$[\![\tau]\!] = v \neq \perp^{[\![\texttt{a}]\!]} \implies \tau \Downarrow_{\texttt{a}} \texttt{v}.$$

4.10 Проблемът за пълна абстракция

Определение 4.7. Денотационната семантика $[\![.]\!]$ се нарича **напълно абстрактна**, ако контекстната и денотационната наредба съвпадат, т.е. за произволни затворени термове τ_1 и τ_2 е изпълнено, че

На англ. *full abstraction*, [9]. [7, стр. 179]

$$\tau_1 \leq_{den} \tau_2 : \mathtt{a} \iff \tau_1 \leq_{ctx} \tau_2 : \mathtt{a}.$$

Теорема 4.7 (Плоткин 1977)**.** Денотационната семантика $[\![.]\!]$ за PCF(bool) **не е** напълно абстрактна.

Да напомним, че от *Теорема* 4.4 винаги имаме следното:

$$\tau \cong_{den} \tau_2 : \mathtt{a} \implies \tau_1 \cong_{ctx} \tau_2 : \mathtt{a}.$$

Тук ще се захванем с търсенето на термове τ_1 и τ_2, за които $\tau_1 \cong \tau_2 : \mathtt{a}$, но $\tau_1 \ncong_{den} \tau_2 : \mathtt{a}$.

Задача 4.9. Да дефинираме функцията $sor : \mathbb{B}_\perp \to (\mathbb{B}_\perp \to \mathbb{B}_\perp)$ по следния начин:

sor идва от sequential or.

sor	$\perp$	$false$	$true$
$\perp$	$\perp$	$\perp$	$\perp$
$false$	$\perp$	$false$	$true$
$true$	$true$	$true$	$true$

Докажете, че sor е определима денотационно в PCF(bool). С други думи, съществува затворен терм σ, за който $[\![\sigma]\!] = sor$.

Упътване. Разгледайте затворения терм

$$\sigma \equiv \lambda \mathtt{x:bool}.\lambda \mathtt{y:bool.if\ x\ then\ true\ else\ y}$$

$\square$

Задача 4.10. Да дефинираме изображението $por : \mathbb{B}_\perp \to (\mathbb{B}_\perp \to \mathbb{B}_\perp)$ по следния начин:

por	$\perp$	$false$	$true$
$\perp$	$\perp$	$\perp$	$true$
$false$	$\perp$	$false$	$true$
$true$	$true$	$true$	$true$

por идва от parallel or.

Докажете, че $por \in [\mathbb{B}_\perp \xrightarrow{\text{н}} [\mathbb{B}_\perp \xrightarrow{\text{н}} \mathbb{B}_\perp]]$ е непрекъснато изображение.

Упътване. Достатъчно е да се съобрази, че $por \in [\mathbb{B}_\perp \xrightarrow{\text{м}} [\mathbb{B}_\perp \xrightarrow{\text{м}} \mathbb{B}_\perp]]$. □

Лема 4.6 (Гордън Плоткин 1977)**.** Изображението por не е определимо в PCF(bool), т.е. не съществува затворен терм τ на езика PCF(bool), за който $[\![\tau]\!] = por$.

Пример 4.4. Да видим, че операторът „или” в хаскел е последователен, а не е паралелен.

```
ghci> True || undefined
True
ghci> undefined || True
*** Exception: Prelude.undefined
```

Задача 4.11. Да разгледаме произволно изображение $f \in [\mathbb{B}_\perp \xrightarrow{\text{н}} [\mathbb{B}_\perp \xrightarrow{\text{н}} \mathbb{B}_\perp]]$, за което имаме ограниченията:

f	$\perp$	$false$	$true$
$\perp$	?	?	$true$
$false$	?	$false$	?
$true$	$true$	?	?

Докажете, че $f = por$. Заключете, че f не е определимо денотационно в PCF(bool).

Упътване. Използвайте монотонността на f. □

В следващите твърдения ще използваме типовете

$$\mathtt{a} \stackrel{\text{деф}}{=} \mathtt{bool} \to (\mathtt{bool} \to \mathtt{bool})$$
$$\mathtt{c} \stackrel{\text{деф}}{=} (\mathtt{bool} \to (\mathtt{bool} \to \mathtt{bool})) \to \mathtt{bool}.$$

За $n = 0, 1$, нека дефинираме затворените термове

Да напомним, че

$$\Omega_{\mathtt{a}} \stackrel{\text{деф}}{=} \mathtt{fix}(\lambda\, \mathtt{x} : \mathtt{a}\,.\,\mathtt{x}).$$

$$\begin{aligned}\Theta_n \equiv \lambda \mathtt{f}{:}\mathtt{a}.\,&\mathtt{if}\ (\mathtt{f}\ \mathtt{true}\ \Omega_{\mathtt{bool}})\ \mathtt{then}\\ &\quad \mathtt{if}\ (\mathtt{f}\ \Omega_{\mathtt{bool}}\ \mathtt{true})\ \mathtt{then}\\ &\qquad \mathtt{if}\ (\mathtt{f}\ \mathtt{false}\ \mathtt{false})\ \mathtt{then}\ \Omega_{\mathtt{bool}}\\ &\qquad\quad \mathtt{else}\ \mathtt{b}_n\\ &\qquad \mathtt{else}\ \Omega_{\mathtt{bool}}\\ &\quad \mathtt{else}\ \Omega_{\mathtt{bool}}\end{aligned}$$

Тук сме положили $\mathtt{b}_0 \stackrel{\text{деф}}{=} \mathtt{true}$ и $\mathtt{b}_1 \stackrel{\text{деф}}{=} \mathtt{false}$.

Задача 4.12. Докажете, че $\emptyset \vdash \Theta_0 : \mathtt{c}$ и $\emptyset \vdash \Theta_1 : \mathtt{c}$.

Твърдение 4.18. $\Theta_i \cong_{ctx} \Omega_{\mathtt{c}} : \mathtt{c}$ за $i = 0, 1$.

Доказателство. Понеже $\mathtt{c} = \mathtt{a} \to \mathtt{bool}$, от (2) на *Твърдение* 4.14 следва, че за да докажем, че $\Theta_i \cong_{ctx} \Omega_{\mathtt{c}} : \mathtt{c}$ е достатъчно да докажем, че

$$(\forall \rho \in \mathtt{PCF}_{\mathtt{a}})[\ \Theta_i\rho \cong_{ctx} \Omega_{\mathtt{c}}\rho : \mathtt{bool}\ ].$$

Оттук, според (2) на *Следствие* 4.6, трябва да докажем, че

$$(\forall \rho \in \mathtt{PCF}_{\mathtt{a}})[\ [\![\Theta_i\rho]\!] = [\![\Omega_{\mathtt{c}}\rho]\!]\ ],$$

което е еквивалентно на

$$(\forall \rho \in \mathtt{PCF}_{\mathtt{a}})[\ [\![\Theta_i]\!]([\![\rho]\!]) = [\![\Omega_{\mathtt{c}}]\!]([\![\rho]\!]) = \bot\ ].$$

Лесно се съобразява, че понеже $[\![\Omega_{\mathtt{bool}}]\!] = \bot$, за да бъде $[\![\Theta_i]\!]([\![\rho]\!]) \neq \bot$, за някое $i = 0, 1$, то трябва да са изпълнени следните три свойства:

- $[\![\rho]\!](true)(\bot) = true$;
- $[\![\rho]\!](\bot)(true) = true$;
- $[\![\rho]\!](false)(false) = false$.

Но тогава според *Задача* 4.11 следва, че $[\![\rho]\!] = por$, което е противоречие, защото знаем, че изображението por не е определимо денотационно в езика `PCF(bool)`. Заключаваме, че за $i = 0, 1$,

$$(\forall \rho \in \mathtt{PCF}_{\mathtt{a}})[\ [\![\Theta_i]\!]([\![\rho]\!]) = \bot = [\![\Omega_{\mathtt{c}}]\!]([\![\rho]\!])\ ].$$

Това означава, че за $i = 0, 1$, $\Theta_i \cong_{ctx} \Omega_{\mathtt{c}} : \mathtt{c}$. □

Сега вече можем да видим защо денотационната семантика на езика `PCF(bool)` не е напълно абстрактна.

Твърдение 4.19. За по-горе дефинираните термове Θ_0 и Θ_1 е изпълнено, че: $\Theta_0 \cong_{ctx} \Theta_1 : \mathtt{c}$, но $\Theta_0 \ncong_{den} \Theta_1 : \mathtt{c}$.

Доказателство. Понеже за $i = 0, 1$ имаме, че $\Theta_i \cong_{ctx} \Omega_{\mathtt{c}} : \mathtt{c}$, от транзитивността на релацията следва, че $\Theta_0 \cong_{ctx} \Theta_1 : \mathtt{c}$.

От друга страна,

$$[\![\Theta_0]\!](por) = true \neq false = [\![\Theta_1]\!](por),$$

откъдето е ясно, че $\Theta_0 \ncong_{den} \Theta_1 : \mathtt{c}$. □

Задача от Кеймбридж 2021 г.

Задача 4.13. Нека разгледаме произволен затворен терм τ на езика PCF(bool) от [5] тип a. Докажете или опровергайте импликациите:

(1) $[\![\tau]\!] = \bot^{[\![\mathtt{a}]\!]} \implies \tau \cong_{ctx} \Omega_{\mathtt{a}} : \mathtt{a}$;

(2) $[\![\tau]\!] = \bot^{[\![\mathtt{a}]\!]} \implies \tau \not\Downarrow_{\mathtt{a}}$;

(3) $\tau \cong_{ctx} \Omega_{\mathtt{a}} : \mathtt{a} \implies \tau \not\Downarrow_{\mathtt{a}}$;

(4) $\tau \not\Downarrow_{\mathtt{a}} \implies \tau \cong_{ctx} \Omega_{\mathtt{a}} : \mathtt{a}$;

(5) $\tau \cong_{ctx} \Omega_{\mathtt{a}} : \mathtt{a} \implies [\![\tau]\!] = \bot^{[\![\mathtt{a}]\!]}$.

Упътване.

(1) Да, защото $[\![\tau]\!] = [\![\Omega_{\mathtt{a}}]\!]$ и оттук $\tau \cong_{ctx} \Omega_{\mathtt{a}} : \mathtt{a}$ според (2) на *Теорема* 4.4.

(2) Тази импликация е вярна за $\mathtt{a} = \mathtt{nat}$ както знаем от *Следствие* **??**. За по-високи типове импликацията не е вярна, защото можем да вземем $\tau \equiv \Omega'_{\mathtt{nat}}$. Тогава $[\![\tau]\!] = \bot^{[\![\mathtt{nat}\to\mathtt{nat}]\!]}$, но $\tau \Downarrow_{\mathtt{nat}\to\mathtt{nat}} \tau$.

(3) Не, защото отново можем да вземем $\tau \equiv \Omega'_{\mathtt{nat}}$. Тогава $\tau \cong_{ctx} \Omega_{\mathtt{nat}\to\mathtt{nat}} : \mathtt{nat} \to \mathtt{nat}$ според *Твърдение* 4.19, но $\tau \Downarrow_{\mathtt{nat}\to\mathtt{nat}} \tau$. Тази импликация е вярна само за $\mathtt{a} = \mathtt{nat}$.

(4) Да, защото, щом $\tau \not\Downarrow_{\mathtt{a}}$, то имаме, че $\tau \cong_{op} \Omega_{\mathtt{a}} : \mathtt{a}$ и според (2) на *Теорема* 4.4 получаваме, че $\tau \cong_{ctx} \Omega_{\mathtt{a}} : \mathtt{a}$.

(5) Накратко, не. По-подробно, очевидно е, че импликацията е изпълнена за $\mathtt{a} = \mathtt{nat}$ или $\mathtt{a} = \mathtt{bool}$. Според *Задача* 4.5, то тази импликация е изпълнена и за типове $\mathtt{a} = \mathtt{a}_1 \to \mathtt{a}_2 \cdots \to \mathtt{a}_n$, където $\mathtt{a}_i = \mathtt{nat}$ или $\mathtt{a}_i = \mathtt{bool}$.

Но нека да вземем $\mathtt{a} = \mathtt{c} = (\mathtt{bool} \to (\mathtt{bool} \to \mathtt{bool})) \to \mathtt{bool}$ както погоре и за Θ_i имаме, че $\Theta_i \cong_{ctx} \Omega_{\mathtt{c}} : \mathtt{c}$. Ако допуснем, че $[\![\Theta_i]\!] = \bot^{[\![\mathtt{c}]\!]}$, то ще следва, че $[\![\Theta_0]\!] = [\![\Theta_1]\!]$, което е противоречие с *Твърдение* 4.19.

□

Така видяхме, че ако вземем всички възможни непрекъснати изображения между две области на Скот, то по този начин губим възможността да изучим точно функциите описващи последователни изчисления. Както видяхме с изображението por, то е непрекъснато, но е очевидно, че за да можем да го дефинираме на нашия език, т.е. да съществува терм, чиято денотационна семантика е por, то трябва да имаме начин за описание на паралелни изчисления в нашия език. Оттук нататък можем да подходим по два начина.

- Можем да се опитаме да разгледаме по-фини версии на понятията непрекъснатост между области на Скот, така че да изключим функции от вида на por.

- Можем да обогатим езика с нови конструкции, които да позволят функции като por да бъдат дефинирани. Такъв пример е езикът PCF(por).

4.10.1 Езикът `PCF(por)`

Дефинираме езика `PCF(por)`като разширение на `PCF(bool)`по следния начин:

- Типовете на `PCF(por)`са типовете на `PCF(bool)`.
- Термовете на `PCF(por)`са термовете на `PCF(bool)`с едно ново правило:

$$\tau ::= \ldots \mid \mathtt{por}(\tau_1, \tau_2).$$

- Имаме следните нови правила в операционната семантика:

$$\frac{\tau_1 \Downarrow_{\mathtt{bool}} \mathtt{true}}{\mathtt{por}(\tau_1, \tau_2) \Downarrow_{\mathtt{bool}} \mathtt{true}} \qquad \frac{\tau_2 \Downarrow_{\mathtt{bool}} \mathtt{true}}{\mathtt{por}(\tau_1, \tau_2) \Downarrow_{\mathtt{bool}} \mathtt{true}}$$

$$\frac{\tau_1 \Downarrow_{\mathtt{bool}} \mathtt{false} \qquad \tau_2 \Downarrow_{\mathtt{bool}} \mathtt{false}}{\mathtt{por}(\tau_1, \tau_2) \Downarrow_{\mathtt{bool}} \mathtt{false}}$$

- Денотационна семантика:

$$[\![\mathtt{por}(\tau_1, \tau_2)]\!]_\Gamma(\overline{u}) \stackrel{\text{деф}}{=} por([\![\tau_1]\!]_\Gamma(\overline{u}), [\![\tau_2]\!](\overline{u})),$$

  където $por \in [\mathcal{B}_\perp^2 \stackrel{\text{н}}{\to} \mathcal{B}_\perp]$ и

$$por(b_1, b_2) \stackrel{\text{деф}}{=} \begin{cases} \mathtt{true}, & \text{ако } b_1 = \mathtt{true} \text{ или } b_2 = \mathtt{true} \\ \mathtt{false}, & \text{ако } b_1 = \mathtt{false} \text{ и } b_2 = \mathtt{false} \\ \perp, & \text{иначе.} \end{cases}$$

Теорема 4.8 (Плоткин 1977)**.** Денотационната семантика $[\![.]\!]$ за езика `PCF(por)`е напълно абстрактна.

[7, стр. 188]

Задача 4.14. Да разгледаме терма

$$\begin{aligned} \tau \equiv\ & \lambda \mathtt{f:(nat->bool)->bool.} \\ & \quad \lambda \mathtt{p:nat->bool.} \\ & \qquad \mathtt{por(p(0),\ f(}\lambda \mathtt{n:nat.p(n+1))).} \end{aligned}$$

Намерете $[\![\mathtt{fix}(\tau)]\!]$.

Кеймбридж 2015 г.

Упътване. Покажете, че $[\![\mathtt{fix}(\tau)]\!](p) = \begin{cases} true, & (\exists n \in \mathbb{N})[p(n) = true] \\ \perp, & \text{иначе.} \end{cases}$ □

Глава 5

Езикът LAMBDA

Термовете τ на езика LAMBDA имат много проста абстрактна граматика:

$$\tau ::= x \mid \tau_1\tau_2 \mid \lambda x.\tau_1.$$

Както преди, когато пишем $\tau[x_1, \ldots, x_n]$ ще имаме предвид, че свободните променливи в τ са измежду $x_1, \ldots, x_n$.

Аналогично, както преди, дефинираме операция субституция.

5.1 Операционна семантика

Full β-reduction

$$\frac{}{(\lambda x.\tau_1)\tau_2 \to \tau_1[x/\tau_2]}$$

$$\frac{\tau_1 \to_\beta \tau_1'}{\tau_1\tau_2 \to \tau_1'\tau_2}$$

$$\frac{\tau_2 \to \tau_2'}{\tau_1\tau_2 \to \tau_1\tau_2'}$$

$$\frac{\tau \to \tau'}{\lambda x.\tau \to \lambda x.\tau'}$$

Определение 5.1. Един терм τ е в **нормална форма**, ако не съществува τ', за който $\tau \to \tau'$.

За един терм τ казваме, че има нормална форма, ако съществува терм τ' в нормална форма, за който $\tau \to^\star \tau'$.

Пример 5.1. Нека $\omega = \lambda x.x\ x$ и $\Omega \stackrel{\text{деф}}{=} \omega\omega$. Получаваме, че

$$\Omega \to_\beta \Omega,$$

което означава, че Ω няма нормална форма.

Макар и релацията $\to$ да определя недетерминистично изчисление, то имаме следното важно свойство.

Теорема 5.1 (Чърч-Росер). Нека $\tau \to^\star \tau_1$ и $\tau \to^\star \tau_2$. Тогава съществува τ', за който $\tau_1 \to^\star \tau'$ и $\tau_2 \to^\star \tau'$.

Следствие 5.1. Нека $\tau \to^\star \tau_1$, $\tau \to^\star \tau_2$ и τ_1 и τ_2 са в нормална форма. Тогава $\tau_1 = \tau_2$.

Ще бъде удобно да дефинираме $\equiv$ като симетричното затваряне на $\to^\star$, т.е.

$$\tau_1 \equiv \tau_2 \iff (\exists \tau_3)[\tau_1 \to^\star \tau_3 \ \&\ \tau_2 \to^\star \tau_3].$$

5.2 Кодиране на аритметиката

Булеви стойности

$$\texttt{true} \overset{\text{деф}}{=} \lambda t.\lambda f.t$$
$$\texttt{false} \overset{\text{деф}}{=} \lambda t.\lambda f.f$$

Булеви операции

За да дефинираме операцията отрицание, интуитивно, трябва да обърнем аргументите.

$$\texttt{not} \overset{\text{деф}}{=} \lambda b.\lambda t.\lambda f.b\ f\ t.$$

Аналогично можем да дефинираме следния терм

$$\texttt{if} \overset{\text{деф}}{=} \lambda b.\lambda t.\lambda f.b\ t\ f.$$

За да се сетим как да дефинираме операцията конюнкция, нека първо да разгледаме следното:

$$\begin{aligned}
&\texttt{true}\,\texttt{true}\ x \equiv \texttt{true}\\
&\texttt{true}\,\texttt{false}\ x \equiv \texttt{false}\\
&\texttt{false}\,\texttt{true}\ x \equiv x\\
&\texttt{false}\,\texttt{false}\ x \equiv x.
\end{aligned}$$

Сега съобразете, че можем да дефинираме следния терм:

$$\texttt{and} \overset{\text{деф}}{=} \lambda b_1.\lambda b_2.b_1\ b_2\ \texttt{false},$$

или така:

$$\texttt{and} \overset{\text{деф}}{=} \lambda b_1.\lambda b_2.b_1\ b_2\ b_1.$$

Да видим сега как можем да изразим операцията дизюнкция.

$$\begin{aligned}
&\texttt{true}\ x\ \texttt{true} &&\equiv x\\
&\texttt{true}\ x\ \texttt{false} &&\equiv x\\
&\texttt{false}\ x\ \texttt{true} &&\equiv \texttt{true}\\
&\texttt{false}\ x\ \texttt{false} &&\equiv \texttt{false}.
\end{aligned}$$

Тогава можем да дефинираме терма

$$\texttt{or} \overset{\text{деф}}{=} \lambda b_1.\lambda b_2.b_1\ \texttt{true}\ b_2,$$

или

$$\texttt{or} \overset{\text{деф}}{=} \lambda b_1.\lambda b_2.b_1\ b_1\ b_2,$$

5.2.1 Естествени числа

Да започнем с дефинирането на затворените $\ulcorner n \urcorner$ за всяко естествено число n.

- $\ulcorner 0 \urcorner \stackrel{\text{деф}}{=} \lambda s.\lambda z.z$
- $\ulcorner 1 \urcorner \stackrel{\text{деф}}{=} \lambda s.\lambda z.s\ z$
- $\ulcorner 2 \urcorner \stackrel{\text{деф}}{=} \lambda s.\lambda z.s\ (s\ z)$
- Оттук се вижда как може да се кодира произволно число n.
- $\ulcorner n \urcorner \stackrel{\text{деф}}{=} \lambda s.\lambda z.s^n\ z$

Понеже имаме, че $\ulcorner n+1 \urcorner = \lambda s.\lambda z.s\ (s^n\ z)$, то е ясно как можем да дефинираме опрацията наследник:

$$\texttt{succ} \stackrel{\text{деф}}{=} \lambda x.\lambda s.\lambda z.s\ (x\ s\ z),$$

защото:

$$\begin{aligned}\texttt{succ}\ \ulcorner n \urcorner &\to^\star \lambda s.\lambda z.s\ (\ulcorner n \urcorner\ s\ z)\\ &\to^\star \lambda s.\lambda z.s\ (s^n\ z)\\ &\to^\star \ulcorner n+1 \urcorner.\end{aligned}$$

5.2.2 Събиране

$$\texttt{plus} \stackrel{\text{деф}}{=} \lambda x.\lambda y.\lambda s.\lambda z.x\ s\ (y\ s\ z),$$

защото:

$$\begin{aligned}\texttt{plus}\ \ulcorner m \urcorner\ \ulcorner n \urcorner &\to^\star \lambda s.\lambda z.\ulcorner m \urcorner\ s\ (\ulcorner n \urcorner\ s\ z)\\ &\to^\star \lambda s.\lambda z.s^m\ (s^n\ z)\\ &\to^\star \ulcorner m+n \urcorner\end{aligned}$$

5.2.3 Умножение

Да започнем от следното наблюдение:

$$\begin{aligned}\texttt{plus}\ \ulcorner m \urcorner &\to^\star \lambda y.\lambda s.\lambda z.\ulcorner m \urcorner\ s\ (y\ s\ z)\\ &\to^\star \lambda y.\lambda s.\lambda z.s^m\ (y\ s\ z).\end{aligned}$$

Тогава:

$$\begin{aligned}(\texttt{plus}\ \ulcorner m \urcorner)^2\ \ulcorner 0 \urcorner &\to^\star (\texttt{plus}\ \ulcorner m \urcorner)\ ((\texttt{plus}\ \ulcorner m \urcorner)\ \ulcorner 0 \urcorner)\\ &\to^\star \texttt{plus}\ \ulcorner m \urcorner\ \ulcorner m \urcorner\\ &\to^\star \ulcorner 2m \urcorner.\end{aligned}$$

Задача 5.1. Докажете, че за произволни естествени числа m и n е изпълнено следното:

$$(\texttt{plus}\ulcorner m\urcorner)^n\ulcorner 0\urcorner \to^\star \ulcorner n*m\urcorner.$$

Тогава можем да дефинираме терма `mult` по следния начин:

$$\texttt{mult} \stackrel{\text{деф}}{=} \lambda x.\lambda y.x\ (\texttt{plus}\ y)\ulcorner 0\urcorner.$$

Лесно се проверява, че:

$$\begin{aligned}\texttt{mult}\ulcorner m\urcorner\ulcorner n\urcorner &\to^\star (\texttt{plus}\ulcorner n\urcorner)\ulcorner 0\urcorner\\ &\to^\star (\texttt{plus}\ulcorner n\urcorner)^m\ulcorner 0\urcorner\\ &\to^\star \ulcorner m*n\urcorner.\end{aligned}$$

За дефинирането на операцията умножение можем да подходим и по друг начин, който ще ни бъде полезен и по-нататък. Да започнем с един пример:

$$\begin{aligned}\ulcorner 2\urcorner(\ulcorner 3\urcorner\ s) &\to^\star (\lambda f.\lambda z.f(f(z)))(\ulcorner 3\urcorner\ s)\\ &\to^\star \lambda z.(\ulcorner 3\urcorner\ s)(\ulcorner 3\urcorner\ s\ z)\\ &\to^\star \lambda z.\ulcorner 3\urcorner\ s\ (s^3\ z)\\ &\to^\star \lambda z.s^3\ (s^3\ z)\\ &\to^\star \lambda z.s^6\ z.\end{aligned}$$

Оттук следва, че:

$$\lambda s.\ulcorner 2\urcorner\ (\ulcorner 3\urcorner\ s) \to^\star \ulcorner 6\urcorner.$$

За произволни затворени термове f и g, да въведем следното означение:

$$f\circ g \stackrel{\text{деф}}{=} \lambda x.g\ (f\ x).$$

Тогава:

$$\ulcorner 2\urcorner\circ\ulcorner 3\urcorner \to^\star \ulcorner 6\urcorner.$$

Задача 5.2. Докажете, че за произволни естествени числа m и n е изпълнено следното:

$$\ulcorner m\urcorner\circ\ulcorner n\urcorner \to^\star \ulcorner m*n\urcorner.$$

Оттук получаваме, че можем да дадем следната дефиниция:

$$\texttt{mult} \stackrel{\text{деф}}{=} \lambda x.\lambda y.x\circ y.$$

5.2.4 Степенуване

Да започнем с един пример:

$$\begin{aligned}\ulcorner 3\urcorner\ulcorner 2\urcorner &\to^\star (\lambda s.\lambda z.s^3\ z)\ulcorner 2\urcorner\\ &\to^\star \lambda z.\ulcorner 2\urcorner^3\ z\\ &\to^\star \lambda z.\ulcorner 2\urcorner\ (\ulcorner 2\urcorner\ (\ulcorner 2\urcorner\ z))\\ &\to^\star \ulcorner 2\urcorner\circ(\ulcorner 2\urcorner\circ\ulcorner 2\urcorner)\\ &\to^\star \ulcorner 2^3\urcorner.\end{aligned}$$

Задача 5.3. Докажете, че за произволни естествени числа m и n е изпълнено следното:

$$\ulcorner m \urcorner \, \ulcorner n \urcorner \to^\star \ulcorner n^m \urcorner.$$

Оттук заключаваме, че можем да дефинираме затворения терм

$$\texttt{mult} \stackrel{\text{деф}}{=} \lambda x.\lambda y.x\ y.$$

Проверка за нула

Искаме да дефинираме затворен терм `iszero` със следните свойства:

$$\begin{aligned}&\texttt{iszero}\,\ulcorner 0 \urcorner \to^\star \texttt{true}\\ &\texttt{iszero}\,\ulcorner n \urcorner \to^\star \texttt{false}, \quad \text{ако } n > 0.\end{aligned}$$

Задача 5.4. Докажете, че за $n > 0$ е изпълнено следното:

$$(\texttt{true}\ a)^n\ b \to^\star a.$$

Оттук можем да съобразим, че:

$$\begin{aligned}&\ulcorner 0 \urcorner\,(\texttt{true}\ a)\ b \to^\star b\\ &\ulcorner n \urcorner\,(\texttt{true}\ a)\ b \to^\star a, \quad \text{ако } n > 0.\end{aligned}$$

Тогава дефинираме

$$\texttt{iszero} \stackrel{\text{деф}}{=} \lambda x.x\ (\texttt{true}\ \texttt{false})\ \texttt{true}.$$

Възможно е да дефимираме терма и така:

$$\texttt{iszero} \stackrel{\text{деф}}{=} \lambda x.x\ (\lambda y.\texttt{false})\ \texttt{true}.$$

5.2.5 Наредени двойки

$$\begin{aligned}&\texttt{pair} \stackrel{\text{деф}}{=} \lambda f.\lambda s.\lambda p.p\ f\ s\\ &\texttt{fst} \stackrel{\text{деф}}{=} \lambda p.p\ \texttt{true}\\ &\texttt{snd} \stackrel{\text{деф}}{=} \lambda p.p\ \texttt{false}\end{aligned}$$

Свойствата, които трябва да проверим са следните:

$$\begin{aligned}&\texttt{fst}\ (\texttt{pair}\ \tau_1\ \tau_2) \to^\star \tau_1\\ &\texttt{snd}\ (\texttt{pair}\ \tau_1\ \tau_2) \to^\star \tau_2.\end{aligned}$$

5.2.6 Изваждане

Първо трябва да дефинираме операция, която работи върху двойки така:

$$(\,\cdot\,, \ulcorner n \urcorner) \mapsto (\ulcorner n \urcorner, \ulcorner n+1 \urcorner).$$

$$\mathtt{pnext} \overset{\text{деф}}{=} \lambda p.\mathtt{pair}\ (\mathtt{snd}\ p)\ (\mathtt{succ}(\mathtt{snd}\ p))$$

Сега можем да дефинираме операцията предшественик, за която са изпълнени свойствата:

$$\mathtt{pred}\,\ulcorner 0 \urcorner \to^{\star} \ulcorner 0 \urcorner$$
$$\mathtt{pred}\,\ulcorner n+1 \urcorner \to^{\star} \ulcorner n \urcorner.$$

Нека първо дефинираме

$$\mathtt{pzero} \overset{\text{деф}}{=} \mathtt{pair}\,\ulcorner 0 \urcorner\,\ulcorner 0 \urcorner.$$

Тогава:

$$\mathtt{pred} \overset{\text{деф}}{=} \lambda n.\mathtt{fst}\ (n\ \mathtt{pnext}\ \mathtt{pzero}\ ).$$

Сега вече можем да дефинираме терм `minus` със свойствата:

$$\mathtt{minus}\,\ulcorner m \urcorner\,\ulcorner n \urcorner \to^{\star} \begin{cases} \ulcorner 0 \urcorner, & \text{ако } m < n \\ \ulcorner m - n \urcorner, & \text{ако } m \geq n. \end{cases}$$

$$\mathtt{minus} \overset{\text{деф}}{=} \lambda m.\lambda n.n\ \mathtt{pred}\ m.$$

Нека да дефинираме и терм, който проверява дали два нумерала на Чърч представляват едно и също число.

Задача 5.5. Дефинирайте терм `cequal` със свойствата:

$$\mathtt{equal}\,\ulcorner m \urcorner\,\ulcorner n \urcorner \to^{\star} \begin{cases} \mathtt{true}, & \text{ако } m = n \\ \mathtt{false}, & \text{иначе}. \end{cases}$$

Упътване. Използвайте, че

$$m = n \iff m \leq n \,\&\, n \leq m \iff m \dot{-} n = 0 \,\&\, n \dot{-} m = 0.$$

□

5.3 Денотационна семантика

Тук следваме [3, глава 18].

- Тук ще разглеждаме областта на Скот $\mathcal{P}(\mathbb{N}) = (\mathcal{P}(\mathbb{N}), \subseteq, \emptyset)$.
- Ще използваме означението D_v за крайното множоество с код v, където, ако $D_v = \{n_0 < n_1 < \cdots < n_{k-1}\}$, то $v = \sum_{i<k} 2^{n_i}$. Ясно е, че на всяко естено число съответства единствено крайно множество и всяко крайно множество има единствен код. Например, на числото 0 съответства множеството $\emptyset$, т.е. $D_0 = \emptyset$. На числото 13 съответства множеството $\{0, 2, 3\}$, т.е. $D_{13} = \{0, 2, 3\}$.

Определение 5.2. Едно изображение $f \in [\mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathbb{N})]$ е **компактно** точно тогава, когато за всяко множество $A \subseteq \mathbb{N}$,

$$x \in f(A) \iff (\exists v)[x \in f(D_v) \;\&\; D_v \subseteq A].$$

Да видим защо това свойство е важно за нас.

Лема 5.1 (Характеризация на непрекъснатите множества)**.** За произволно $f \in [\mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathbb{N})]$, следните условия са еквивалентни.

(1) $f \in [\mathcal{P}(\mathbb{N}) \overset{\text{н}}{\to} \mathcal{P}(\mathbb{N})]$;

(2) $f \in [\mathcal{P}(\mathbb{N}) \overset{\text{м}}{\to} \mathcal{P}(\mathbb{N})]$ и f е компактно.

Доказателство. $(1) \implies (2)$. Ясно е, че щом f е непрекъснато, то f е монотонно. Да видим защо f е компактно.

Първо, нека $D_v \subseteq A$ и $x \in f(D_v)$. Щом f е монотонно, то е ясно, че $x \in f(A)$.

Второ, ясно е, че за произволно множество A можем да разгледаме веригата от крайни множества $A_i = \{x \in A : x < i\}$, за която $A = \bigcup_i A_i$. Щом $f \in [\mathcal{P}(\mathbb{N}) \overset{\text{н}}{\to} \mathcal{P}(\mathbb{N})]$, то $f(\bigcup_i A_i) = \bigcup_i f(A_i)$. Тогава, ако $x \in f(A)$, то $x \in f(A_i)$ за някое i.

$(2) \implies (1)$. Нека A_i е верига от множества. Трябва да проверим, че

$$f(\bigcup_i A_i) = \bigcup_i f(A_i).$$

Първо, нека $x \in \bigcup_i f(A_i)$. Тогава $x \in f(A_i)$ за някое i. От монотонността на f имаме, че $x \in f(\bigcup_i A_i)$.

Второ, нека $x \in f(\bigcup_i A_i)$. От компактността на f следва, че $x \in f(D_v)$, където $D_v \subseteq \bigcup_i A_i$. Щом D_v е крайно и $(A_i)_{i=0}^{\infty}$ е верига, то съществува i_0, за което $D_v \subseteq A_{i_0}$. От монотонността на f следва, че $x \in f(A_{i_0})$ и оттук $x \in \bigcup_i f(A_i)$. $\square$

Денотационната семантика на всеки затворен терм на езика LAMBDA ще бъде просто едно множество от ествени числа. Проблемът е, че в някои ситуации, това множество трябва да се интерпретира като функция, а в други случаи трябва да се интерпретира като входни данни. За да формализираме тази идея, ще въведем две функции.

Определение 5.3. За произволно изображение $f \in [\mathcal{P}(\mathbb{N}) \overset{\text{н}}{\to} \mathcal{P}(\mathbb{N})]$,

$$\mathtt{graph}(f) \overset{\text{деф}}{=} \{\langle x, v\rangle \mid x \in f(D_v)\}.$$

Дефинираме изображение $\mathtt{fun} \in [\mathcal{P}(\mathbb{N}) \to [\mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathbb{N})]]$, където

$$\mathtt{fun}(A)(B) \overset{\text{деф}}{=} \{x \mid (\exists v)[\langle x, v\rangle \in A \;\&\; D_v \subseteq B]\}.$$

Преди да престъпим кам дефиницията на денотационната семантика, да видим някои основни свойства на току-що въведените функции.

Задача 5.6. Докажете, че $\mathtt{graph} \in [[\mathcal{P}(\mathbb{N}) \overset{\text{н}}{\to} \mathcal{P}(\mathbb{N})] \overset{\text{н}}{\to} \mathcal{P}(\mathbb{N})]$.

Задача 5.7. Докажете, че $\mathtt{fun} \in [\mathcal{P}(\mathbb{N}) \overset{\text{н}}{\to} [\mathcal{P}(\mathbb{N}) \overset{\text{н}}{\to} \mathcal{P}(\mathbb{N})]]$.

Задача 5.8. Докажете, че $\mathtt{fun}(\mathtt{graph}(f)) = f$.

Задача 5.9. Докажете, че за всяко $A \subseteq \mathbb{N}$ е изпълнено, че $A \subseteq \mathtt{graph}(\mathtt{fun}(A))$.

Определение 5.4. Нека $\tau[x_1, \dots, x_n]$ е терм в езика LAMBDA. Дефинираме $[\![\tau]\!] \in [\mathcal{P}(\mathbb{N})^n \to \mathcal{P}(\mathbb{N})]$ по следния начин.

- $[\![x_i]\!](\bar{u}) = u_i$;
- $[\![\tau_1\tau_2]\!](\bar{u}) = \mathtt{fun}([\![\tau_1]\!](\bar{u}))([\![\tau_2]\!](\bar{u}))$;
- $[\![\lambda y.\tau]\!](\bar{u}) = \mathtt{graph}(\mathtt{curry}([\![\tau]\!])(\bar{u}))$.

Лема 5.2. За всеки терм $\tau[x_1, \dots, x_n]$ е изпълнено, че

$$[\![\tau]\!] \in [\mathcal{P}(\mathbb{N})^n \overset{\text{н}}{\to} \mathcal{P}(\mathbb{N})].$$

Доказателство. Доказателството е директно с индукция по построението на термовете.

- $[\![x_i]\!] = \texttt{proj}^n_i$, където $\texttt{proj}^n_i(\bar{u}) = u_i$. Знаем, че $\texttt{proj}^n_i$ е непрекъснато изображение.
- $[\![\tau_1\tau_2]\!] = \texttt{uncurry}(\texttt{fun}) \circ ([\![\tau_1]\!] \times [\![\tau_2]\!])$;
- $[\![\lambda y.\tau]\!] = \texttt{graph} \circ \texttt{curry}([\![\tau]\!])$.

□

Естествено е да видим дали операцията за замяна на променлива с терм е съвместима с нашата денотационна семантика.

Лема 5.3 (Лема за замяната)**.** За произволни термове $\tau[x_1, \dots, x_n, y]$ и $\delta[x_1, \dots, x_n]$ е изпълнено, че

$$[\![\tau[y/\delta]]\!](\bar{u}) = [\![\tau]\!](\bar{u}, [\![\delta]\!](\bar{u})).$$

Доказателство.

- Случаят, когато $\tau = x_i$ ще оставим на читателя.
- Нека $\tau = y$. Тогава е ясно, че $[\![y[y/\delta]]\!](\bar{u}) = [\![\delta]\!](\bar{u}) = [\![y]\!](\bar{u}, [\![\delta]\!](\bar{u}))$.
- Нека $\tau = \tau_1\tau_2$. Този случай не крие изненади:

$$\begin{aligned}
[\![(\tau_1\tau_2)[y/\delta]]\!](\bar{u}) &= [\![\tau_1[y/\delta]\tau_2[y/\delta]]\!](\bar{u}) \\
&= \texttt{fun}([\![\tau_1[y/\delta]]\!](\bar{u}))([\![\tau_2[y/\delta]]\!](\bar{u})) \\
&= \texttt{fun}([\![\tau_1]\!](\bar{u}, [\![\delta]\!](\bar{u}))([\![\tau_2]\!](\bar{u}, [\![\delta]\!](\bar{u}))) && \text{// от } \textbf{(И.П.)} \\
&= [\![\tau_1\tau_2]\!](\bar{u}, [\![\delta]\!](\bar{u})).
\end{aligned}$$

- Нека $\tau = \lambda z.\mu$, където $\mu[x_1, \dots, x_n, z, y]$. Тогава от **(И.П.)** за $\delta[x_1, \dots, x_n]$ имаме, че за произволни множества $\bar{u}, v$ е изпълнено следното:

$$[\![\mu[y/\delta]]\!](\bar{u}, v) = [\![\mu]\!](\bar{u}, v, [\![\delta]\!](\bar{u})).$$

  Без ограничение, можем да запишем горното равенство и така:

$$[\![\mu[y/\delta]]\!](\bar{u}, v) = [\![\mu]\!](\bar{u}, [\![\delta]\!](\bar{u}), v),$$

  откъдето следва, че

$$\texttt{curry}([\![\mu[y/\delta]]\!])(\bar{u}) = \texttt{curry}([\![\mu]\!])(\bar{u}, [\![\delta]\!](\bar{u})). \tag{5.1}$$

  Сега вече сме готови да завършим доказателството:

$$\begin{aligned}
[\![(\lambda z.\mu)[y/\delta]]\!](\bar{u}) &= [\![\lambda z.\mu[y/\delta]]\!](\bar{u}) \\
&= \texttt{graph}(\texttt{curry}(\mu[y/\delta])(\bar{u})) \\
&= \texttt{graph}(\texttt{curry}(\mu)(\bar{u}, [\![\delta]\!](\bar{u}))) && \text{// от (5.1)} \\
&= [\![\lambda z.\mu]\!](\bar{u}, [\![\delta]\!](\bar{u})).
\end{aligned}$$

□

Теорема 5.2 (Коректност)**.** За всеки два терма τ_1 и τ_2

$$\tau_1 \to^{\star} \tau_2 \implies [\![\tau_1]\!] = [\![\tau_2]\!].$$

Следствие 5.2. За всеки два терма τ_1 и τ_2

$$\tau_1 \equiv \tau_2 \implies [\![\tau_1]\!] = [\![\tau_2]\!].$$

5.4 Рекурсия

Нека отново да положим $\omega \stackrel{\text{деф}}{=} \lambda x.xx$ и $\Omega \stackrel{\text{деф}}{=} \omega\omega$. Вече сме виждали терма Ω, който е пример за терм без нормална форма. Погледнато по друг начин, можем да си мислим, че ако се опитаме да изчислим Ω то влизаме в бездънна рекурсия. Да видим как можем да развием това наблюдение.

Първо, да разгледаме сега $\omega \stackrel{\text{деф}}{=} \lambda f.\lambda x.ffx$. Тогава за $\Omega \stackrel{\text{деф}}{=} \omega\omega$, получаваме:

$$\Omega\tau \to \Omega\tau.$$

Нека сега

$$\omega \stackrel{\text{деф}}{=} \lambda f.\lambda x.\mathtt{if}\ x\ \mathtt{then}\ x\ \mathtt{else}\ f\ f\ x.$$

Тогава за $\Omega \stackrel{\text{деф}}{=} \omega\omega$ получаваме

$$\begin{aligned} \Omega\ \mathtt{true} &\to \mathtt{if}\ \mathtt{true}\ \mathtt{then}\ \mathtt{true}\ \mathtt{else}\ \omega\ \omega\ \mathtt{true} \\ &\to \mathtt{true} \\ \Omega\ \mathtt{false} &\to \mathtt{if}\ \mathtt{false}\ \mathtt{then}\ \mathtt{false}\ \mathtt{else}\ \omega\ \omega\ \mathtt{false} \\ &\to \underbrace{\omega\ \omega\ \mathtt{false}}_{\Omega\ \mathtt{false}} \end{aligned}$$

Използвайки тази идея, можем вече да дефинираме рекурсивни функции. Нека отново да разгледаме рекурсивната дефиниция функцията факториел:

$$\mathtt{fact} \stackrel{\text{деф}}{=} \lambda x.\mathtt{if}\ \mathtt{iszero}(x)\ \mathtt{then}\ \ulcorner 1 \urcorner\ \mathtt{else}\ \mathtt{mult(x,\ fact(pred(x)))}.$$

Вече трябва да е интуитивно ясно, че би трябвало да можем да представим $fact$ като най-малката неподвижна точка на терма

$$F \stackrel{\text{деф}}{=} \lambda f.\lambda x.\mathtt{if}\ \mathtt{iszero}(x)\ \mathtt{then}\ \ulcorner 1 \urcorner\ \mathtt{else}\ \mathtt{mult}(x, f(\mathtt{pred}(x))).$$

Сега да разгледаме терма

$$F' \stackrel{\text{деф}}{=} \lambda f.\lambda x.\mathtt{if}\ \mathtt{iszero}(x)\ \mathtt{then}\ \ulcorner 1 \urcorner\ \mathtt{else}\ \mathtt{mult}(x, ff(\mathtt{pred}(x))).$$

Да проверим, че имаме следното свойство:

Задача 5.10. Докажете, че за всяко естествено число е изпълнено свойството:

$$F'\ F'\ n \to^\star n!.$$

Доказателство. Проверката е директна с индукция по n: Нека $n = 0$.

$$\begin{aligned} F'\ F'\ 0 &\to^\star \mathtt{if}\ 0 == 0\ \mathtt{then}\ 1\ \mathtt{else}\ n * (F'F'(n-1)) \\ &\to^\star 1. \end{aligned}$$

Нека сега $n > 0$.

$$\begin{aligned} F'\ F'\ n &\to^{\star} \texttt{if}\ n == 0\ \texttt{then}\ 1\ \texttt{else}\ n * (F'F'(n-1)) \\ &\to^{\star} n * (F'F'(n-1)) \\ &\to^{\star} n * (n-1)! && \text{// от } \textbf{(И.П.)} \\ &\to^{\star} n! \end{aligned}$$

□

Сега да видим, че можем да получим по *равномерен* начин F' от F. Това означава, че съществува терм ρ, за който $F' = \rho F$. Но това е много лесно. Понеже

$$\lambda g.F(gg) \to^{\star} F',$$

то нека положим

$$\rho \stackrel{\text{деф}}{=} \lambda f.(\lambda g.f(gg)).$$

Ясно е, че

$$\rho F \to^{\star} F'.$$

Заключаваме, че:

$$\begin{aligned} fact &\equiv F'F' \\ &\equiv (\rho F)(\rho F) \\ &\equiv (\lambda f.(\lambda g.f(gg))(\lambda g.f(gg)))F \end{aligned}$$

Да положим

$$Y \stackrel{\text{деф}}{=} (\lambda f.(\lambda g.f(gg))(\lambda g.f(gg))).$$

Получаваме, че

$$fact \equiv YF.$$

Понеже знаем, че

$$Ffact \equiv fact,$$

откъдето следва, че

$$F(YF) \equiv YF,$$

заключаваме, че термът Y ни дава *равномерен* начин да намираме най-малката неподвижна точка на произволен затворен терм τ, т.е.

$$\tau(Y\tau) \equiv Y\tau.$$

Наистина:

$$\begin{aligned} Y\tau &\equiv \overbrace{(\lambda x.\tau(x\ x))}^{\tau'}\overbrace{(\lambda x.\tau(x\ x))}^{\tau'} \\ &\equiv \tau'\tau' \\ &\equiv \tau(\tau'\tau') \\ &\equiv \tau(Y\tau). \end{aligned}$$

Един проблем с Y комбинатора е, че нямаме $Y\tau \to^\star \tau(Y\tau)$.

Ясно е, че Y ни дава начин да намираме неподвижна точка на терма τ. Дали получаваме най-малката неподвижна точка?

Теорема 5.3. Нека τ е затворен терм. Тогава

$$[\![Y\tau]\!] = \mathtt{lfp}(\mathtt{fun}([\![\tau]\!])).$$

Доказателство. Нека положим $A_0 \stackrel{\text{деф}}{=} [\![Y\tau]\!]$. Първо, защо A_0 е неподвижна точка на $\mathtt{fun}([\![\tau]\!])$, т.е. защо

$$[\![Y\tau]\!] = \mathtt{fun}([\![\tau]\!])([\![Y\tau]\!])?$$

Но това е лесно. От дефиницията на денотационната семантика следва, че

$$\mathtt{fun}([\![\tau]\!])([\![Y\tau]\!]) = [\![\tau(Y\tau)]\!].$$

Вече знаем, че $Y\tau \equiv \tau(Y\tau)$, то от *Следствие* 5.2 следва, че $[\![Y\tau]\!] = [\![\tau(Y\tau)]\!]$.

Сега към по-интересната част. Нека A е произволна неподвижна точка, т.е. $A = \mathtt{fun}([\![\tau]\!])(A)$. Трябва да докажем, че $[\![Y\tau]\!] \subseteq A$. Понеже $Y\tau \equiv \tau'\tau'$, то имаме следното:

$$\begin{aligned}[\![Y\tau]\!] &= [\![\tau'\tau']\!]\\ &= \mathtt{fun}([\![\tau']\!])([\![\tau']\!])\\ &= \bigcup_{D_v \subseteq [\![\tau']\!]} \mathtt{fun}(D_v)(D_v).\end{aligned}$$

Това означава, че трябва да докажем, че за всяко крайно множество $D_v \subseteq [\![\tau]\!]$, $\mathtt{fun}(D_v)(D_v) \subseteq A$.

За да докажем това, първо трябва да докажем едно помощно твърдение.

Твърдение 5.1. Нека $D_v \subseteq [\![\tau']\!]$. Тогава

$$\mathtt{fun}(D_v)(D_v) \subseteq \bigcup_{u<v} \mathtt{fun}([\![\tau]\!])(\mathtt{fun}(D_u)(D_u)).$$

Доказателство. Нека $x \in \mathtt{fun}(D_v)(D_v)$. Това означава, че съществува u, за което $\langle u, x\rangle \in D_v$, $D_u \subseteq D_v$ и $D_v \subseteq [\![\tau']\!]$. Ясно е, че $u < v$. Сега, щом $\langle u, x\rangle \in D_v$, понеже $D_v \subseteq [\![\tau']\!]$, то

$$\begin{aligned}\langle u, x\rangle \in [\![\tau']\!] &\implies \langle u, x\rangle \in [\![\lambda x.\tau(xx)]\!] && // \ \tau' \stackrel{\text{деф}}{=} \lambda x.\tau(xx)\\ &\implies \langle u, x\rangle \in \mathtt{graph}(\mathtt{curry}([\![\tau(xx)]\!]))\end{aligned}$$

$$\begin{aligned}
&\Longrightarrow\ x \in [\![\tau(xx)]\!](D_u)\\
&\Longrightarrow\ x \in \mathtt{fun}([\![\tau]\!])([\![xx]\!](D_u))\\
&\Longrightarrow\ x \in \mathtt{fun}([\![\tau]\!])(\mathtt{fun}([\![x]\!](D_u))([\![x]\!](D_u))\\
&\Longrightarrow\ x \in \mathtt{fun}([\![\tau]\!])(\mathtt{fun}(D_u)(D_u)).
\end{aligned}$$

□

Твърдение 5.2. Нека $D_v \subseteq [\![\tau']\!]$. Тогава $\mathtt{fun}(D_v)(D_v) \subseteq A$.

Доказателство. Индукция по v.

Нека $v = 0$. Ясно е, че $D_0 = \emptyset$. Тогава $\mathtt{fun}(\emptyset)(\emptyset) = \emptyset \subseteq A$.

Нека $v > 0$. Нека $x \in \mathtt{fun}(D_v)(D_v)$. Това означава, че съществува $u < v$, за което $x \in \mathtt{fun}([\![\tau]\!])(\mathtt{fun}(D_u)(D_u))$. Понеже $D_u \subseteq D_v$, от **(И.П.)** следва, че $\mathtt{fun}(D_u)(D_u) \subseteq A$. Тогава от монотонността на $\mathtt{fun}([\![\tau]\!])$, получаваме, че $x \in \mathtt{fun}([\![\tau]\!])(A) = A$. □

Заключаваме, че

$$\begin{aligned}
[\![Y\tau]\!] &= \bigcup_{D_v \subseteq [\![\tau']\!]} \mathtt{fun}(D_v)(D_v)\\
&\subseteq A,
\end{aligned}$$

което означава, че $[\![Y\tau]\!]$ е най-малката неподвижна точка на $[\![\tau]\!]$. □

Обикновено в литературата затворените термове се наричат комбинатори.

Комбинатор на Тюринг

Сега да тръгнем от другата страна. Искаме да намерим терм Θ, за който за всеки затворен терм τ е изпълнено:

$$\Theta\tau \equiv \tau(\Theta\tau). \tag{5.2}$$

С други думи, искаме

$$\Theta \equiv \lambda g.g(\Theta g).$$

Това означава, че за терма

$$F \stackrel{\text{деф}}{=} \lambda f.\lambda g.g(fg),$$

термът Θ трябва да е решение на уравнението $\Theta \equiv F\Theta$. Отново, използваме трикът от по-горе и разглеждаме

$$F' \stackrel{\text{деф}}{=} \lambda f.\lambda g.(gffg) \equiv \rho F,$$

където

$$\rho \stackrel{\text{деф}}{=} \lambda f.(\lambda g.f(gg)).$$

Тогава $\Theta \stackrel{\text{деф}}{=} F'F'$ е решение на уравнението 5.2.

$$\Theta \stackrel{\text{деф}}{=} (\lambda f.\lambda g.g(ffg))(\lambda f.\lambda g.g(ffg)).$$

За разлика от Y комбинатора, комбинаторът Θ има следното хубаво свойство:

$$\Theta\tau \stackrel{\text{деф}}{=} F'F'\tau \to^\star \tau(F'F'\tau) = \tau(\Theta\tau).$$

Оттук можем да получем безкрайно много термове за неподвижни точки. Например, можем да разгледаме

$$F \stackrel{\text{деф}}{=} \lambda x.\lambda y.\lambda f.f(xyxf).$$

Тогава

$$(FFF)\tau \equiv \tau(FFF\tau).$$

Можем да продължим така:

$$F \stackrel{\text{деф}}{=} \lambda xyzuvw.w(xyuxyuw).$$

Тогава

$$(F^6)\tau \equiv \tau(F^6\tau).$$

Сега да дефинираме терма

$$F \stackrel{\text{деф}}{=} \lambda abcdefghijklmnopqstuvwxyzr.(r(thisisafixedpointcombinator))$$

Тогава

$$FFFFFFFFFFFFFFFFFFFFFFFFFFF$$

е терм за неподвижна точка.

Библиография

[1] S. Abramsky и A. Jung. "Domain Theory". В: *Handbook of Logic in Computer Science*. Под ред. на S. Abramsky, D. M. Gabbay и T. S. E. Maibaum. Т. 3. Clarendon Press, 1994, с. 1—168.

[2] D. N. Arden. "Delayed-logic and finite-state machines". В: *2nd Annual Symposium on Switching Circuit Theory and Logical Design (SWCT 1961)*. 1961, с. 133—151.

[3] Hendrik Barendregt. *The Lambda Calculus: Its Syntax and Semantics*. Т. 103: North-Holland. 1984.

[4] Roberto Bruni и Ugo Montanari. *Models of Computation*. Springer, 2017.

[5] Marcelo Fiore. *Denotational Semantics*. 2021. URL: https://www.cl.cam.ac.uk/teaching/2021/DenotSem/ (дата на посещ. 10.01.2021).

[6] Marcelo Fiore. *Lecture Notes on Denotational Semantics*. https://www.cl.cam.ac.uk/teaching/1920/DenotSem/DenotSemNotes.pdf. Достъпен на 3 януари 2020 г.

[7] Carl Gunter. *Semantics of Programming Languages*. MIT Press, 1992.

[8] Robert Harper. *Practical Foundations for Programming Languages*. CUP, 2016.

[9] Robin Milner. "Fully abstract models of typed λ-calculi". В: *Theoretical Computer Science* 4.1 (1977), с. 1—22. ISSN: 0304-3975. DOI: 10.1016/0304-3975(77)90053-6.

[10] Benjamin Pierce. *Types and Programming Languages*. MIT Press, 2002.

[11] John C. Reynolds. *Theories of Programming Languages*. Cambridge University Press, 1998.

[12] Glynn Winskel. *The Formal Semantics of Programming Languages*. MIT Press, 1993.

[13] Стела Николова и Александра Соскова. *Семантика на езиците за програмиране. Ръководство*. СОФТЕХ, 2008.

[14] Ангел Дичев и Иван Сосков. *Теория на програмите*. СУ „Св. Климент Охридски", 1998.

Азбучен указател

α-еквивалентност, 85
β-правило, 137

Клини, 27
Кнастер, 27
Лема за замяна, 66, 145
Лема за разместването, 16
Плоткин, 131, 135
Скот, 7
Тарски, 27
Теорема на Клини, 27
денотационна семантика, 51
    по име, 66, 71
език
    безконтекстен, 61
завършена програма, 119
изображение
    композиция, 22
    монотонно, 11
    непрекъснато, 13
изоморфизъм, 44
израз, 83
константа, 64
контекст, 119
най-малка неподвижна точка, 27
неподвижна точка, 27
плоска област на Скот, 7
правило на Ардън, 59
преднеподвижна точка, 28
променлива
    нулев тип, 64
    обектова, 64
    функционална, 64

регулярен език, 49
рекурсивна програма, 65
семантика
    денотационна, 143
    операционна, 50, 73, 91, 137
система, 36
    най-малко решение, 36
    решение, 36
стойност, 84
субституция, 86
терм, 64, 85, 136
    стойност, 66
термален оператор, 66
тип, 83, 87
частична наредба, 4