

ЗАДАЧИ ЗА ИЗПИТ ПО λ СТД

(2016/17)

ПРАВИЛА ЗА ОЦЕНЯВАНЕ

- За успешно полагане на изпита са нужни ≥ 15 т., от които:
 - ≥ 6 т. от раздел 1,
 - ≥ 3 т. от раздел 2,
 - ≥ 3 т. от раздел 3.
- При покриване на горните критерии, оценката по шестобална система се пресмята с формулата $\min(6, \frac{p}{5})$, където p е общият брой на събраните точки.
- На изпита ще се очаква да можете да обясните и защитите решенията си.
- На изпита е позволено ползването на записките от лекции, както и на допълнителна литература.
- Възможно е на изпита да ви бъде поставена допълнителна задача за оформяне на крайната оценка.

1. БЕЗТИПОВО λ -СМЯТАНЕ

Дефиниция 1.1 (Субституция). Нека $M, N \in \Lambda$, $x \in V$. Дефинираме индуктивно по построението на M субституцията на x с N в M , която ще отбелязваме с $M[x \mapsto N]$.

- (1) $x[x \mapsto N] := N$
- (2) $y[x \mapsto N] := y$ за $y \neq x$
- (3) $(M_1 M_2)[x \mapsto N] := (M_1[x \mapsto N])(M_2[x \mapsto N])$
- (4) $(\lambda_x P)[x \mapsto N] := \lambda_x P$
- (5) $(\lambda_y P)[x \mapsto N] := \lambda_y(P[x \mapsto N])$ за $y \neq x$ и $(x \notin P$ или $y \notin \text{FV}(N))$ и е недефинирана иначе.

Задача 1.1. (1 т.) Да се дефинира формално с индукция операция “преименуване на свързана променлива”, която по даден терм $M \in \Lambda$, променлива $x \in V$ и променлива $y \notin \text{FV}(M) \cup \text{BV}(M)$, дефинира нов терм M_y^x , който представлява резултата от заменянето на всички свързани срещания на x в M с y .

Задача 1.2. (2 т.) Да се покаже, че операцията субституция може да се разглежда като тотална с точност до релацията $\stackrel{\alpha}{=}$. Формално, нека считаме че са дадени произволни $x \in V$ и $N \in \Lambda$. Да се покажат следните две свойства:

- (1) За всяко $M \in \Lambda$, съществува $M' \stackrel{\alpha}{=} M$, така че $M'[x \mapsto N]$ е дефинирана.

Дата: 13 юни 2017 г.

- (2) Ако $M \stackrel{\alpha}{=} M' \in \Lambda$, $N \stackrel{\alpha}{=} N' \in \Lambda$ и $M[x \mapsto N]$ и $M'[x \mapsto N']$ са дефинирани едновременно, то $M[x \mapsto N] \stackrel{\alpha}{=} M'[x \mapsto N']$.

Задача 1.3. (2 т.) Да разгледаме алтернативна тотална дефиниция на операцията субституция $M[x \rightsquigarrow N]$, която е дефинирана по същия начин като $M[x \mapsto N]$, с изключение на условието (5), което е променено по следния начин:

- (5) $(\lambda_y P)[x \rightsquigarrow N] := \lambda_y(P[x \rightsquigarrow N])$ за $y \neq x$

Казваме, че операцията субституция е коректна, ако $FV(N) \cap BV(M) = \emptyset$. Да се покаже, че:

- (1) ако $M[x \rightsquigarrow N]$ е коректна, то $M[x \mapsto N]$ е дефинирана $M[x \mapsto N] \equiv M[x \rightsquigarrow N]$;
- (2) има случай, в който $M[x \mapsto N]$ е дефинирана, но $M[x \rightsquigarrow N]$ не е коректна;
- (3) за всяко $M \in \Lambda$ можем да намерим $M' \stackrel{\alpha}{=} M$, така че $M'[x \rightsquigarrow N]$ да е коректна.

Задача 1.4. (3 т.) Да се направи програмна реализация на операцията субституция, която при нужда преименува свързаните променливи по подходящ начин при прилагане, за да осигури коректност.

Задача 1.5. (5 т.) Да се дефинира формално понятието “граф, съответстващ на λ -терм” и да се докаже, че два λ -терма са α -еквивалентни тогава и само тогава, когато съответните им графи са изоморфни.

Дефиниция 1.2 (Безименни термове, Λ_n , Λ^*). С едновременна индукция за всички $n \in \mathbb{N}$ дефинираме множествата Λ_n от безименни термове с не повече от n различни свободни променливи.

- (1) $i \in \Lambda_n$ за всяко естествено число i , за което $0 \leq i < n$.
- (2) Ако $M, N \in \Lambda_n$, то $(MN) \in \Lambda_n$ и апликацията на M над N .
- (3) Ако $M \in \Lambda_{n+1}$, то $(\lambda M) \in \Lambda_n$ и абстракцията над променливата с индекс 0 в M .

С $\Lambda^* := \bigcup_{n=0}^{\infty} \Lambda_n$ отбелязваме множеството на всички безименни λ -термове.

Задача 1.6. (1 т.) Да се докаже, че за $m < n$ е вярно, че $\Lambda_m \subseteq \Lambda_n$.

Дефиниция 1.3. Нека $X \subseteq V$ е множество от променливи. Дефинираме $\Lambda_X := \{M \in \Lambda : FV(M) \subseteq X\}$.

Задача 1.7. (5 т.) Да се дефинират фамилията от изображения $\#_{\Gamma} : \Lambda_{|\Gamma|} \rightarrow \Lambda_{\text{dom } \Gamma}$ и $\flat_{\Gamma} : \Lambda_{\text{dom } \Gamma} \rightarrow \Lambda_{|\Gamma|}$ за даден контекст от имена $\Gamma := x_{n-1}, \dots, x_0$, които извършват превод между термове с имена и безименни термове. Да се покаже, че

- (1) $\#_{\Gamma}(\flat_{\Gamma}(M)) \stackrel{\alpha}{=} M$ за всеки терм $M \in \Lambda_{\text{dom } \Gamma}$
- (2) $\flat_{\Gamma}(\#_{\Gamma}(M)) \equiv M$ за всеки терм $M \in \Lambda_{|\Gamma|}$

Задача 1.8. (5 т.) Да се реализира програма, която позволява въвеждането и извеждането на λ -термове в два формата: с имена (Λ) и без имена (Λ^*) на променливите. За преобразуването между двата формата да се използва автоматично генериран контекст от имена от вида $\Gamma := x_{n-1}, \dots, x_0$.

Дефиниция 1.4 (Изместване). Дефинираме $\uparrow_c^d(M) : \Lambda_n \rightarrow \Lambda_{n+d}$ с индукция по построението на терма $M \in \Lambda_n$.

- (1) $\uparrow_c^d(k) := \begin{cases} k, & \text{за } 0 \leq k < c, \\ k + d, & \text{за } c \leq k < n. \end{cases}$
- (2) $\uparrow_c^d(MN) := (\uparrow_c^d(M))(\uparrow_c^d(N))$
- (3) $\uparrow_c^d(\lambda M) := \lambda \uparrow_{c+1}^d(M)$

Дефинираме $\uparrow^d(M) := \uparrow_0^d(M)$.

Дефиниция 1.5 (Субституция на безименни термове). Нека $M, N \in \Lambda_n$ и $k \in \mathbb{N}$. С индукция по M дефинираме субституцията $M[k \mapsto N] \in \Lambda_n$.

- (1) $k[k \mapsto N] := N$
- (2) $i[k \mapsto N] := i$ за $i \neq k$
- (3) $(M_1 M_2)[k \mapsto N] := (M_1[k \mapsto N])(M_2[k \mapsto N])$
- (4) $(\lambda M)[k \mapsto N] := \lambda(M[k + 1 \mapsto \uparrow^1(N)])$

Задача 1.9. (5 т.) Да се провери, че двете дефиниции за субституции са съгласувани относно изображенията $\#_\Gamma$ и $\flat_\Gamma$. За целта, нека фиксираме контекст от имена $\Gamma := x_{n-1}, \dots, x_0$. Да се покаже, че

- (1) $\#_\Gamma(M)[x_i \mapsto \#_\Gamma(N)] \stackrel{\alpha}{=} \#_\Gamma(M[i \mapsto N])$ за произволни $M, N \in \Lambda_n$,
- (2) $\flat_\Gamma(M)[i \mapsto \flat_\Gamma(N)] \equiv \flat_\Gamma(M[x_i \mapsto N])$ за произволни $M, N \in \Lambda_{\text{dom } \Gamma}$.

Задача 1.10. (3 т.) Да се направи програмна реализация на субституция над безименни термове.

Дефиниция 1.6 (Подтерм). Дефинираме индуктивно функцията $\text{Sub} : \Lambda \Rightarrow 2^\Lambda$, където $\text{Sub}(M)$ е множеството на подтермовете на M индуктивно:

- (1) $\text{Sub}(x) := \{x\}$
- (2) $\text{Sub}(MN) := \text{Sub}(M) \cup \text{Sub}(N) \cup \{MN\}$
- (3) $\text{Sub}(\lambda_x M) := \text{Sub}(M) \cup \{\lambda_x M\}$

Дефинираме релацията “ M е подтерм на N ” по следния начин: $M \leq N := M \in \text{Sub}(N)$.

Задача 1.11. (2 т.) Докажете, че релацията $\leq$ е частична наредба, т.е. че е рефлексивна, транзитивна и антисиметрична.

Дефиниция 1.7 (λ -затваряне). Нека е дадена бинарна релация над λ -термове $R \subseteq \Lambda^2$. Дефинираме индуктивно релацията R^λ , която наричаме λ -затваряне на R , по следния начин:

- (1) Ако $(M, N) \in R$, то $(M, N) \in R^\lambda$.
- (2) Ако $(M, N) \in R^\lambda$, $P \in \Lambda$ и $x \in V$, то
 - $(MP, NP) \in R^\lambda$,
 - $(PM, PN) \in R^\lambda$,
 - $(\lambda_x M, \lambda_x N) \in R^\lambda$.

Ако $R^\lambda = R$, казваме че R е λ -съвместима.

Интуитивно, два терма M и N са в релация R^λ ако те съвпадат синтактично с изключение на два техни съответни подтерма $M' \subseteq M$ и $N' \subseteq N$, които са в релация R .

Задача 1.12. (1 т.) Дайте друг пример за λ -съвместима релация.

Задача 1.13. (1 т.) Докажете, че R^λ е λ -съвместима за произволна релация R .

Дефиниция 1.8 (λ -контекст). λ -контекст наричаме λ -терм, в който има точно едно срещане на специална променлива, която ще наричаме “дупка” и ще означаваме с $\square$. Формално можем да дефинираме λ -контексти индуктивно по следния начин:

- (1) $\square$ е λ -контекст
- (2) Ако E е λ -контекст, а M е λ -term, то (ME) и (EM) са λ -контексти
- (3) Ако E е λ -контекст, а x е произволна променлива, то $\lambda_x E$ е λ -контекст

Заместване на λ -терм M в контекст E дефинираме като субституция на дупката $\square$ в контекста E с конкретния терм M . Такова заместване ще отбелязваме с $E[M]$, което всъщност ще съответства на $E[\square \mapsto M]$. При такова заместване ще се откажем от конвенцията, която забранява прихващането на свободните променливи и ще позволим това да се случва. С индукция по контекста E лесно се вижда, че $M \leq E[M]$ за произволен терм M .

Задача 1.14. (2 т.) Да се докаже, че $(M, N) \in R^\lambda$ тогава и само тогава, когато съществуват λ -контекст E , и два подтерма $M' \leq M$ и $N' \leq N$, така че

- (1) $E[M'] \equiv M$
- (2) $E[N'] \equiv N$
- (3) $(M', N') \in R$.

Интуитивно, това свойство изразява факта, че два терма са в релация R^λ тогава и само тогава, когато те се различават само на едно място в структурата си, и на това място съответните термове са в релация R .

Задача 1.15. (3 т.) Да се дефинира формално релацията $\xrightarrow{\beta}$ за множеството безименните термове Λ^* и да се докаже, че двете β -редукции са съгласувани, т.е. за произволен контекст от имена Γ

- (1) $\vdash_\Gamma(M) \xrightarrow{\beta} \vdash_\Gamma(N)$, ако $M, N \in \Lambda$, $FV(MN) \subseteq \text{dom } \Gamma$ и $M \xrightarrow{\beta} N$.
- (2) $\#_\Gamma(M) \xrightarrow{\beta} P$, ако $M, N \in \Lambda_{|\Gamma|}$, $M \xrightarrow{\beta} N$ и $P \equiv \#_\Gamma(N)$.

Дефиниция 1.9 (Апликативни термове). Дефинираме множеството от апликативни λ -термове $A\Lambda \subseteq \Lambda$ индуктивно по следния начин

- (1) Ако $x \in V$, то $x \in A\Lambda$.
- (2) Ако $M, N \in A\Lambda$, то $MN \in A\Lambda$.

Задача 1.16. (3 т.) Нека k и s са две фиксирани променливи от V . Да се дефинира изображение $\Phi : \Lambda \Rightarrow A\Lambda$, което превежда произволен λ -терм в апликативен терм (превод на λ -смятане в комбинаторна логика), така че за произволно $M \in \Lambda$:

- (1) $FV(\Phi(M)) = FV(M) \cup \{k, s\}$ и
- (2) $M \xrightarrow{\beta} \Phi(M)[k \mapsto K][s \mapsto S]$.

Екстра кредит: (2 т.) Да се направи програмна реализация на изображението Φ .

Дефиниция 1.10 (екстенционално равенство). Казваме, че M и N са екстенционално равни и бележим $\lambda + \text{ext} \models M = N$, ако

- (1) $M \xrightarrow{\beta} N$,
- (2) за произволно $x \notin FV(MN)$ е вярно, че $\lambda + \text{ext} \models Mx = Nx$.

Задача 1.17. (3 т.) Да се докаже, че релацията $\lambda + \text{ext} \models M = N$ е λ -съвместима релация на еквивалентност.

Задача 1.18. Да се дефинират следните комбинатори и с помощта на индукция да се докаже формално тяхната коректност:

- (1 т.) c_S , такъв че $c_S c_n \stackrel{\beta}{=} c_{n+1}$ за $n \in \mathbb{N}$.
- (2 т.) c_+ , такъв че $c_+ c_m c_n \stackrel{\beta}{=} c_{m+n}$ за $m, n \in \mathbb{N}$.
- (2 т.) c_* , такъв че $c_* c_m c_n \stackrel{\beta}{=} c_{mn}$ за $m, n \in \mathbb{N}$.
- (2 т.) c_{exp} , такъв че $c_{\text{exp}} c_m c_n \stackrel{\beta}{=} c_{m^n}$ за $m, n \in \mathbb{N}$.
- (3 т.) c_{hyp} , такъв че $c_{\text{hyp}} c_m c_n \stackrel{\beta}{=} c_p$, където $p = \underbrace{m^{\dots^m}}_n$ за $m, n \in \mathbb{N}$.

Задача 1.19. (1 т.) Нека са дадени следните дефиниции:

- $c'_+ := \lambda_{m,n,f,x} m f(n f x)$
- $c''_+ := \lambda_{m,n} m c_S n$

Да се покажат термове M и N , за които $c'_+ M N \stackrel{\beta}{\neq} c''_+ M N$.

Задача 1.20. (1 т.) Нека дефинираме $c_I := \lambda_n n c_S c_0$.

- (1) Да се докаже, че за произволно $n \in \mathbb{N}$ е изпълнено $c_I c_n \stackrel{\beta}{=} c_n$.
- (2) Вярно ли е, че $c_I \stackrel{\beta\eta}{=} I$? Да се докаже или да се покаже контрапример.

Задача 1.21. (2 т.) Нека дефинираме

$$\begin{aligned} c_{\text{tt}} &:= \lambda_{x,y} x \\ c_{\text{ff}} &:= \lambda_{x,y} y \\ c_{\langle \rangle} &:= \lambda_{x,y,z} z x y \\ c_{\perp} &:= \lambda_p p c_{\text{tt}} \\ c_{\lrcorner} &:= \lambda_p p c_{\text{ff}} \\ c_P &:= \lambda_n c_{\lrcorner} (n (\lambda_z c_{\langle \rangle} (c_S (c_{\perp} z)) (c_{\lrcorner} z)) (c_{\langle \rangle} c_0 c_0)). \end{aligned}$$

Да се докаже, че $c_P c_0 \stackrel{\beta}{=} c_0$ и $c_P c_n \stackrel{\beta}{=} c_{n-1}$ за $n > 0$.

Задача 1.22. (2 т.) Нека дефинираме

$$c_{\dagger} := \lambda_n c_{\lrcorner} (n (\lambda_z c_{\langle \rangle} (c_S (c_{\perp} z)) (c_* (c_S (c_{\perp} z)) (c_{\lrcorner} z))) (c_{\langle \rangle} c_0 c_1)).$$

Да се докаже, че $c_{\dagger} c_n = c_{n!}$ за произволно $n \in \mathbb{N}$.

Задача 1.23. (2 т.) Да се дефинират комбинатори $c_{=}$ и $c_{<}$, за които за произволни $m, n \in \mathbb{N}$:

- $c_{=} c_m c_n \stackrel{\beta}{=} c_{m=n}$
- $c_{<} c_m c_n \stackrel{\beta}{=} c_{m < n}$.

Да се докаже формално коректността на дефинираните комбинатори.

Екстра кредит: (1 т.) Да се направи програмна реализация.

Задача 1.24. (2 т.) Да се дефинират комбинатори c_{quot} и c_{rem} , така че за произволни $m, n \in \mathbb{N}$:

- $c_+ (c_* (c_{\text{quot}} c_m c_n) c_n) (c_{\text{rem}} c_m c_n) \stackrel{\beta}{=} c_m$,
- $c_{<} (c_{\text{rem}} c_m c_n) c_k \stackrel{\beta}{=} c_{\text{tt}}$.

Да се докаже формално коректността на дефинираните комбинатори.

Екстра кредит: (1 т.) Да се направи програмна реализация.

Задача 1.25. (3 т.) Да се дефинират комбинатори $c_!$ и c_{prime} , така че за произволни $m, n \in \mathbb{N}$:

- $c_! c_m c_n \stackrel{\beta}{=} c_{\exists k(km=n)}$;
- $c_{\text{prime}} c_n \stackrel{\beta}{=} c_{\neg \exists k, l > 1 (kl=n)}$.

Да се докаже формално коректността на дефинираните комбинатори.

Екстра кредит: (1 т.) Да се направи програмна реализация.

Задача 1.26. (1 т.) Да се дефинира комбинатор c_- , така че за произволни $m, n \in \mathbb{N}$ е изпълнено, че $c_- c_m c_n \stackrel{\beta}{=} c_{m \dot{-} n}$, където $m \dot{-} n := \max(m - n, 0)$. Да се докаже формално коректността на дефинирания комбинатор.

Екстра кредит: (1 т.) Да се направи програмна реализация.

Задача 1.27. (5 т.) Да се предложи дефиниция на списъци в безтиповото λ -смятане. Да се реализират комбинатори реализиращи стандартните функции *map*, *foldr* и *filter*.

Екстра кредит: (3 т.) Да се направи програмна реализация.

Задача 1.28. (2 т.) Да се дефинира комбинатор A , който реализира функцията на Акерман, т.е. за който

- $A c_0 c_n \stackrel{\beta}{=} c_{n+1}$,
- $A c_{m+1} c_0 \stackrel{\beta}{=} A c_m c_1$,
- $A c_{m+1} c_{n+1} \stackrel{\beta}{=} A c_m (A c_{m+1} c_n)$.

Екстра кредит: (1 т.) Да се направи програмна реализация. Да се докаже формално

Задача 1.29. (5 т.) Да се дефинира комбинатор M , който симулира операцията “минимизация”, т.е. ако t е комбинатор, за който съществува число n , такова че

- (1) $tc_n \stackrel{\beta}{=} c_0$
- (2) $\forall m < n \exists k (tc_m \stackrel{\beta}{=} c_{k+1})$,

то $Mt \stackrel{\beta}{=} c_n$. Да се докаже формално коректността на дефинирания комбинатор M .

Екстра кредит: (2 т.) Да се направи програмна реализация.

Дефиниция 1.11 (решимост). Казваме, че един затворен терм M е *решим*, ако съществува число $n \in \mathbb{N}$ и термове N_i за $1 \leq i \leq n$, така че $MN_1 \dots N_n \stackrel{\beta}{=} I$.

Дефиниция 1.12 (λ -определимост). Нека $f : \mathbb{N}^n \rightarrow \mathbb{N}$ е частична функция над естествените числа. Казваме, че f е λ -*определима*, ако съществува комбинатор F такъв, че за всяка n -торка числа $x_1, \dots, x_n$ имаме:

- (1) ако $f(x_1, \dots, x_n)$ е дефинирана и има стойност y , то $Fc_{x_1} \dots c_{x_n} \stackrel{\beta}{=} c_y$;
- (2) ако $f(x_1, \dots, x_n)$ не е дефинирана, то $Fc_{x_1} \dots c_{x_n}$ е нерешим.

Дефиниция 1.13 (частично рекурсивни функции). Дефинираме класа на частично рекурсивните функции (ЧРФ) $f : \mathbb{N}^n \rightarrow \mathbb{N}$ с индукция, както следва:

- (1) Функциите $O(x) := 0$ (нула) $S(x) := x+1$ (наследник) и $I_k^n(x_1, \dots, x_n) := x_k$ (проекция) са ЧРФ.
- (2) Ако функциите $f : \mathbb{N}^m \rightarrow \mathbb{N}$ и $g_i : \mathbb{N}^n \rightarrow \mathbb{N}$ за $1 \leq i \leq m$ са ЧРФ, то и функцията $h : \mathbb{N}^n \rightarrow \mathbb{N}$, дефинирана като $h(x_1, \dots, x_n) := f(g_1(x_1, \dots, x_n), \dots, g_m(x_1, \dots, x_n))$, е ЧРФ (композиция).
- (3) Ако функциите $f : \mathbb{N}^n \rightarrow \mathbb{N}$ и $g : \mathbb{N}^{n+2} \rightarrow \mathbb{N}$ са ЧРФ, то и функцията $h : \mathbb{N}^{n+1} \rightarrow \mathbb{N}$, дефинирана с равенствата $h(0, x_1, \dots, x_n) := f(x_1, \dots, x_n)$ и $h(y+1, x_1, \dots, x_n) := g(y, f(y, x_1, \dots, x_n), x_1, \dots, x_n)$ също е ЧРФ (примитивна рекурсия).
- (4) Ако функцията $f : \mathbb{N}^{n+1} \rightarrow \mathbb{N}$ е ЧРФ, то функцията $h : \mathbb{N}^n \rightarrow \mathbb{N}$, дефинирана като $h(x_1, \dots, x_n) := t$, където $f(t, x_1, \dots, x_n) = 0$ и $f(k, x_1, \dots, x_n) < 0$ за $k < t$, също е ЧРФ (минимизация).

Задача 1.30. (8 т.) Да се докаже, че всяка ЧРФ е λ -определима.

Задача 1.31. (21 т.) Да се докаже, че всяка функция, изчислима с машина на Тюринг е λ -определима.

Задача 1.32. (1 т.) Да се покаже пример, че $\xrightarrow{\beta}$ не изпълнява свойството на диаманта.

Задача 1.33. (2 т.) Да се докаже, че ако $M \xrightarrow{\beta} M'$, то $M[x \mapsto N] \xrightarrow{\beta} M'[x \mapsto N]$

Задача 1.34. (3 т.) Да се докаже, че ако $N \xrightarrow{\beta} N'$, то $M[x \mapsto N] \xrightarrow{\beta} M[x \mapsto N']$

Задача 1.35. (5 т.) Да се покаже, че $\xrightarrow{\beta} \subseteq \xrightarrow{1} \subseteq \xrightarrow{\beta}$, т.е. че $M \xrightarrow{\beta} N$ влече $M \xrightarrow{1} N$ и че $M \xrightarrow{1} N$ влече $M \xrightarrow{\beta} N$.

Задача 1.36. (3 т.) Разглеждаме множеството $NF \subseteq \Lambda$ дефинирано индуктивно както следва:

- Ако $x \in V$ и $M_1, \dots, M_n \in NF$, то $xM_1 \dots M_n \in NF$
- Ако $x \in V$ и $M \in NF$, то $\lambda_x M \in NF$.

Да се докаже, че NF е множеството от всички термове в β -нормална форма.

Задача 1.37. (2 т.)

- Да се покаже, че AR и NR са стратегии за пълна редукция.
- Да се покаже пример, че CBV и CBN не са стратегии за пълна редукция.

Задача 1.38. (13 т.) Да се реализира програма, която позволява дефиниране на безтипови λ -термове и изпълняване над тях на четирите стратегии за редукция (апликативна, нормална, извикване по стойност, извикване по име).

2. ТИПОВО λ -СМЯТАНЕ

Дефиниция 2.1 (Ниво на тип). За произволен тип $\tau \in T$ дефинираме индуктивно ниво на типа τ , което бележим с $|\tau|$:

- $|\alpha| := 0$

- $|\rho \Rightarrow \sigma| := \max(|\rho| + 1, |\sigma|)$.

Задача 2.1. (3 т.) Да се докаже, че типовите променливи α не са обитаеми.

Задача 2.2. (2 т.) Да се покаже, че за всяко естествено число n :

- (1) съществува тип σ_n от ниво n , който е обитаем;
- (2) съществува тип τ_n от ниво n , който не е обитаем.

Дефиниция 2.2 (Изтриване на тип). Дефинираме индуктивно изображение $|\cdot| : \Lambda^T \rightarrow \Lambda$, което изобразява типизирани λ -термове в Church стил в съответните безтипови λ -термове като изтрива типа.

- $|x^\tau| := x$,
- $|(M^{\rho \Rightarrow \sigma} N^\rho)^\sigma| := |M^{\rho \Rightarrow \sigma}| |N^\rho|$,
- $|(\lambda_{x^\rho} M^\sigma)^{\rho \Rightarrow \sigma}| := \lambda_x |M^\sigma|$.

Задача 2.3. (3 т.) Да се покаже, че

- (1) за всеки затворен типизиран терм $M^\tau \in \Lambda^T$ може да се намери типов извод на типовото създание $|M^\tau| : \tau$.
- (2) за всеки безтипов терм $M \in \Lambda$, за който имаме типов извод на типовото създание $M : \tau$, съществува типизиран терм N^τ в стил Church, така че $|N| \equiv M$.

Задача 2.4. (8 т.) Да се даде дефиниция на безименни типизирани λ -термове. Именен контекст наричаме списък от променливи. Да се дефинират следните две фамилии от изображения, индексирани по типове τ :

- Φ_τ , което по именен контекст Γ и типизиран λ -терм t^τ , такива че Γ съдържа всички свободни променливи на t получава безименен λ -терм $\Phi_\tau(\Gamma, t^\tau)$ от тип τ , и
- Ψ_τ , което по именен контекст Γ и безименен типизиран λ -терм M^τ получава обикновен λ -терм $\psi_\tau(\Gamma, M^\tau)$ от тип τ със свободни променливи измежду Γ ,

такива че за всеки тип τ е изпълнено, че

- $\Phi_\tau(\Gamma, \Psi_\tau(\Gamma, M^\tau)) = M^\tau$ и
- $\Psi_\tau(\Gamma, \Phi_\tau(\Gamma, t^\tau)) = t^\tau$.

Дефиниция 2.3 (Слабо типизирани термове). Дефинираме множеството на слабо типизирани термове Λ^{WT} индуктивно по следния начин:

- Ако $x \in V$, то $x \in \Lambda^{WT}$,
- Ако $M, N \in \Lambda^{WT}$, то $(MN) \in \Lambda^{WT}$,
- Ако $x \in V$, $\tau \in T$ и $M \in \Lambda^{WT}$, то $(\lambda_{x:\tau} M) \in \Lambda^{WT}$.

Типов извод за слабо типизирани термове се дефинира аналогично на типов извод на безтипови термове, с ограничението, че дърво с корен $(\lambda_{x:\tau} M) : \rho \Rightarrow \sigma$ може да съществува само, ако $\tau \equiv \rho$.

Задача 2.5. (3 т.) Да се покаже, че ако $M \in \Lambda^{WT}$ е слабо типизиран терм и $\Gamma \vdash M : \sigma$ и $\Gamma \vdash M : \tau$, то $\sigma \equiv \tau$.

Задача 2.6. (3 т.) Да се докаже, че ако $\Gamma, x : \rho \vdash M : \sigma$ и $\Gamma \vdash N : \rho$, то $\Gamma \vdash M[x \mapsto N] : \sigma$.

Задача 2.7. (3 т.) Да се докаже, че ако $\Gamma \vdash M : \tau$ и $M \xrightarrow{\beta} N$, то $\Gamma \vdash N : \tau$.

Дефиниция 2.4 (Типова субституция). Типова субституция наричаме всяко изображение $\xi : TV \rightarrow T$. Ако τ е тип, дефинираме индуктивно $\tau\xi$ — прилагането на ξ към τ :

- $\alpha\xi := \xi(\alpha)$,
- $(\rho \Rightarrow \sigma)\xi := (\rho\xi \Rightarrow \sigma\xi)$.

Казваме, че τ е по-общ от σ (отбелязваме $\tau \supseteq \sigma$) ако има субституция ξ , така че $\tau\xi \equiv \sigma$.

Задача 2.8. (2 т.) Да се докаже, че ако $\vdash M : \tau$ и $\tau \supseteq \sigma$, то $\vdash M : \sigma$.

Задача 2.9. (2 т.) Да се докаже, че $\supseteq$ е частична преднаредба, т.е. е рефлексивна и транзитивна релация.

Екстра кредит: (1 т.) Да се покаже пример, че $\supseteq$ не е антисиметрична релация.

Задача 2.10. (1 т.) Да се каже, че за всяко $n \in \mathbb{N}$ и произволен тип τ е вярно, че $\vdash c_n : (\tau \Rightarrow \tau) \Rightarrow \tau \Rightarrow \tau$.

Задача 2.11. (3 т.) Да се докаже, че ако $\Gamma \vdash M : \tau$ и $N \leq M$, тогава $\Gamma \vdash N : \sigma$ за някой тип σ .

Задача 2.12. (2 т.) Да се реализират всички операции за сравнение на естествени числа в системите T или PCF .

Задача 2.13. (3 т.) Да се реализират операциите за частно и остатък от целочислено деление в системите T или PCF .

Задача 2.14. (3 т.) Да се реализират предикатите за проверка за делимост и проверка за простота на число в системите T или PCF .

Задача 2.15. (3 т.) Да се реализира функцията на Акерман в системата T .

Задача 2.16. (8 т.) Да се разшири системата T с тип за полиморфни списъци $L(\rho)$ с елементи от тип ρ , като се дефинират подходящи конструктори, рекурсор, и правила за редукция.

3. ТЕОРИЯ НА ДОКАЗАТЕЛСТВАТА

Задача 3.1. Да се реализира програма, която позволява дефиниране на доказателства в някоя от следните системи:

- **(8 т.)** Хилбертова система $H[mic]$
- **(13 т.)** секвенциално смятане $G[123][mic]$
- **(13 т.)** система за естествен извод $N[mic]$

Възможни са два варианта за реализация:

- (1) доказателствата винаги са коректни по построение;
- (2) позволено е да бъдат построени некоректни доказателства, но е реализирана функция за проверка за коректност.

Задача 3.2. (3 т.) Да се докаже, че ако $G1[mic] \vdash A$, то може да се постори доказателство на A , в което всички аксиоми са атомарни формули, т.е. са от вида $p\vec{t} \Rightarrow p\vec{t}$.

Задача 3.3. Да се докажат законите на *de Morgan* в $G[123]c$ или Nc , а кждето е възможно в $G[123]t$ и Nt , че:

- (1) (1 т.) $\neg(A \wedge B) \leftrightarrow \neg A \vee \neg B$
- (2) (1 т.) $\neg(A \vee B) \leftrightarrow \neg A \wedge \neg B$
- (3) (1 т.) $\neg\forall_x A \leftrightarrow \exists_x \neg A$
- (4) (1 т.) $\neg\exists_x A \leftrightarrow \forall_x \neg A$

Екстра кредит: (по 1 т.) Доказателствата да се опишат в λ -синтаксис или в системите MINLOG или COQ.

Задача 3.4. Да се докаже в $G[123]c$ или Nc , а където е възможно в $G[123]t$ или Nt , че

- (1) (2 т.) $A \tilde{\vee} B \leftrightarrow A \vee B$
- (2) (2 т.) $A \tilde{\wedge} B \leftrightarrow A \wedge B$
- (3) (2 т.) $\tilde{\exists}_x A \leftrightarrow \exists_x A$

Екстра кредит: (по 1 т.) Доказателствата да се опишат в λ -синтаксис или в системите MINLOG или COQ.

Задача 3.5. (3 т.) Да се докажат Хилбертовите аксиоми на Nt в $G[123]t$ или Nt .

Задача 3.6. (3 т.) Да се докаже, че формулите A^g получени с превод на Gödel-Gentzen са стабилни, т.е. да се построят доказателства в $G[123]t$ или Nt на $\neg\neg A^g \rightarrow A^g$.

Задача 3.7. (13 т.) Преводът на Kuroda е трансформация на формули A до A^q , където $A^q := \neg\neg A_q$, а A_q се дефинира индуктивно така:

- $P(\vec{x})_q := P(\vec{x})$
- $(A \rightarrow B)_q := A_q \rightarrow B_q$
- $(A \vee B)_q := A_q \vee B_q$
- $(A \wedge B)_q := A_q \wedge B_q$
- $(\forall_x A)_q := \forall_x \neg\neg A_q$
- $(\exists_x A)_q := \exists_x A_q$

Да се покаже, че

- (1) $\vdash_c A \leftrightarrow A^q$
- (2) $\Gamma \vdash_c A$ тогава и само тогава когато $\Gamma^q \vdash_m A^q$

Задача 3.8. (3 т.) Да се покаже, че за всяка формула A от аксиомите $\text{efq}_p \forall_{\vec{x}} (\perp \rightarrow p\vec{x})$ в Nt е изводима формулата $\perp \rightarrow A$.

Екстра кредит: (1 т.) Да се даде пример за формула A , за която формулата $\neg\neg A \rightarrow A$ не е изводима в Nt от аксиомите $\forall_{\vec{x}} (\neg\neg P(\vec{x}) \rightarrow P(\vec{x}))$ и да се обясни защо.

Задача 3.9. Да се докаже в $G[123]c$ или Nc , че

- (1) (2 т.) $(A \rightarrow \exists_x B) \rightarrow \exists_x (A \rightarrow B)$, ако $x \notin \text{FV}(A)$.
- (2) (2 т.) $((A \rightarrow B) \rightarrow C) \rightarrow (A \rightarrow C) \rightarrow C$
- (3) (2 т.) $(A \rightarrow B) \vee (B \rightarrow A)$

Екстра кредит: (по 1 т.) Доказателствата да се опишат в λ -синтаксис или в системите MINLOG или COQ.

Задача 3.10. Да се докаже, че:

- (2 т.) $Ni \vdash (A \rightarrow B) \tilde{\vee} (A \rightarrow C) \rightarrow A \rightarrow B \vee C$
- (2 т.) $Nm \vdash (A \vee B \rightarrow C) \rightarrow (A \rightarrow C) \wedge (B \rightarrow C)$

- (2 т.) $Ni \vdash (A \rightarrow \tilde{\exists}_x B) \rightarrow \tilde{\exists}_x (A \rightarrow B)$, ако $x \notin FV(A)$
- (2 т.) $Nm \vdash \exists_x (A \rightarrow B) \rightarrow \forall_x A \rightarrow B$, ако $x \notin FV(B)$.

Екстра кредит: (по 1 т.) Доказателствата да се опишат в λ -синтаксис или в системите MINLOG или COQ.

Задача 3.11. (5 т.) Да се докаже, че

- (1) Правилата $\tilde{\vee}_{1,2}^+$, $\tilde{\wedge}^+$ и $\tilde{\exists}^+$ са изводими в $G[123]m$ или Nm .
- (2) Правилата $\tilde{\vee}^-$, $\tilde{\wedge}^-$ и $\tilde{\exists}^-$ са изводими в $G[123]c$ или Nc .

Екстра кредит: (2 т.) Доказателствата да се опишат в λ -синтаксис или в системите MINLOG или COQ.

Задача 3.12. (5 т.) Да се докаже, че β -редукцията в $Nm(\rightarrow, \forall)$ е локално конфлуентна, т.е. ако $M_1 \xrightarrow{\beta} M \xrightarrow{\beta} M_2$, то съществува доказателство N , за което $M_1 \xrightarrow{\beta} N \xleftarrow{\beta} M_2$.

Задача 3.13. (13 т.) Да се реализира програма, която нормализира дадено доказателство в $Nm(\rightarrow, \forall)$ чрез оценяване (NbE).

Задача 3.14. (5 т.) Да се разпишат всички 15 пермутиращи редукции за правилата, породени от аксиомите $\wedge^-$, $\vee^-$, $\exists^-$ над правилата, породени от аксиомите $\rightarrow^-$, $\forall^-$, $\wedge^-$, $\vee^-$, $\exists^-$ в двата записа: дърво на извод и термов синтаксис.

Задача 3.15. (8 т.) Да се напишат доказателства в HA^ω , че операциите от задача 2.12 за сравнение са съответно строга/частична наредба (за $<$, $>$, $\leq$, $\geq$) и релация на еквивалентност (за $=$).

Екстра кредит: (2 т.) Доказателствата да се опишат в λ -синтаксис или в системите MINLOG или COQ.

Задача 3.16. (5 т.) Да се напишат доказателства в HA^ω , които доказват коректността на операциите от задача 2.13 за намиране на частно и остатък при целочислено деление.

Екстра кредит: (2 т.) Доказателствата да се опишат в λ -синтаксис или в системите MINLOG или COQ.

Задача 3.17. (8 т.) Да се напише в HA^ω доказателството на Евклид, че съществуват безкрайно големи прости числа.

Екстра кредит: (2 т.) Доказателствата да се опишат в λ -синтаксис или в системите MINLOG или COQ.