ЗАДАЧИ ЗА ИЗПИТ ПО λСТД

(2015/16)

Правила за оценяване

- За успешно полагане на изпита са нужни $\geq$ 15 т., от които:
  $\geq$ 6 т. от раздел 2,
  $\geq$ 3 т. от раздел 3,
  $\geq$ 3 т. от раздел 4.
- 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. Индукция

Задача 1.1. ***(3 т.)*** *Естествените положителни числа, които нямат прости делители по-големи от 5 се наричат* числа на Хеминг. *Формално, множеството от тези числа може да се дефинира директно, чрез описание на специфичното им свойство:*

$$H_1 := \{h \in \mathbb{N} : \ \textit{ако } p/h \textit{ и } p \textit{ е просто, то } p \in \{2,3,5\}\}.$$

Друга възможна дефиниция е индуктивната:

(1) $1 \in H_2$
(2) *Ако* $h \in H_2$*, то* $2h \in H_2, 3h \in H_2, 5h \in H_2$.

Да се докаже, че $H_1 = H_2$.

Дефиниция 1.1 (Съждителна формула)**.** Нека е дадено изброимо множество от съждителни променливи V. Дефинираме понятието *съждителна формула* индуктивно по следния начин.

(1) Ако $P \in V$, то P е съждителна формула
(2) Ако φ е съждителна формула, то $(\neg\varphi)$ е съждителна формула.
(3) Ако φ и ψ са съждителни формули, то $(\varphi \wedge \psi)$, $(\varphi \vee \psi)$ и $(\varphi \to \psi)$ са съждителни формули.

Задача 1.2. ***(1 т.)*** *Да се докаже, че във всяка съждителна формула има еднакъв брой леви и десни скоби.*

Дата: 6 юни 2016 г.

Дефиниция 2.1 (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 $y \not\equiv x$
(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 $y \not\equiv x$ и $x \notin P$ или $y \notin \mathsf{FV}(N)$

Задача 2.1. ***(2 т.)*** *Да се дефинира формално с индукция операция "преименуване на свързана променлива", която по даден терм $M \in \Lambda$, променлива $x \in \mathsf{BV}(M)$ и променлива $y \notin \mathsf{FV}(M) \cup \mathsf{BV}(M)$, дефинира нов терм M^x_y, който представлява резултата от заменянето на всички* свързани *срещания на x в M с y.*

Задача 2.2. ***(3 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.*
(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 2.3. ***(2 т.)*** *Да разгледаме алтернативна 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* $y \not\equiv x$

Казваме, че операцията субституция е коректна, *ако* $\mathsf{FV}(N) \cap \mathsf{BV}(M) = \emptyset$. *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 2.4. ***(5 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 2.5. ***(1 т.)*** *Да се покажат $M, N, P \in \Lambda$ и $x, y \in V$ такива, че $M[x \mapsto N][y \mapsto P] \not\equiv M[y \mapsto P][x \mapsto N]$.*

Задача 2.6. ***(2 т.)*** *Да се покаже, че съществуват $M, N, P \in \Lambda$, така че $M[x \mapsto N][y \mapsto P] \not\equiv M[y \mapsto P][x \mapsto N[y \mapsto P]]$, ако някое от двете условия $x \notin \mathsf{FV}(P)$ и $x \not\equiv y$ е нарушено.*

Задача 2.7. ***(8 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 2.2 (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 $(MN) \in \Lambda_n$ е апликацията на M над N.
(3) Ако $M \in \Lambda_{n+1}$, то $\lambda M \in \Lambda_n$ е абстракцията над променливата с индекс 0 в M.

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

Дефиниция 2.3. Нека $X \subseteq V$ е множество от променливи. Дефинираме $\Lambda_X := \{M \in \Lambda : \mathsf{FV}(M) \subseteq X\}$.

Задача 2.8. ***(5 т.)*** *Да се дефинират фамилиите от изображения* $\#_\Gamma : \Lambda_{|\Gamma|} \to \Lambda_{\operatorname{dom}\Gamma}$ *и* $\flat_\Gamma : \Lambda_{\operatorname{dom}\Gamma} \to \Lambda_{|\Gamma|}$ *за даден контекст от имена* $\Gamma := x_{n-1}, \ldots, x_0$*, които извършват превод между термове с имена и безименни термове. Да се покаже, че*

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

Задача 2.9. ***(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* $\Gamma := x_{n-1}, \ldots, x_0$.

Дефиниция 2.4 (Изместване)**.** Дефинираме $\uparrow_c^d (M) : \Lambda_n \to \Lambda_{n+d}$ 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 2.5 (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 2.10. ***(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* $\Gamma := x_{n-1}, \ldots, x_0$*. 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* $M, N \in \Lambda_{\operatorname{dom}\Gamma}$.

Задача 2.11. ***(5 т.)*** *Да се направи програмна реализиация на субституция над безименни термове.*

Дефиниция 2.6 (Подтерм)**.** Дефинираме индуктивно релацията "M е подтерм на N", което отбелязваме $M \leq N$ за $M, N \in \Lambda$.

(1) $M \leq M$.

(2) Ако $\lambda_x M \leq N$, то $M \leq N$.
(3) Ако $MN \leq P$, то $M \leq P$ и $N \leq P$.

Задача 2.12. ***(5 т.)*** *Докажете, че релацията $\leq$ е частична наредба, т.е. че е рефлексивна, транзитивна и антисиметрична.*

Дефиниция 2.7 (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 2.13. ***(2 т.)*** *Дайте друг пример за λ-съвместима релация.*

Задача 2.14. ***(2 т.)*** *Докажете, че R^λ е λ-съвместима за произволна релация R.*

Дефиниция 2.8 (λ-контекст)**.** λ-контекст наричаме λ-терм, в който има точно едно срещане на специална променлива, която ще наричаме "дупка" и ще означаваме с $[]$. Формално можем да дефинираме λ-контексти индуктивно по следния начин:

(1) $[]$ 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 $[]$ в контекста E с конкретния терм M. Такова заместване ще отбелязваме с $E[M]$, което всъщност ще съответства на $E[[] \mapsto M]$. При такова заместване ще се откажем от конвенцията, която забранява прихващането на свободните променливи и ще позволим това да се случва.

Задача 2.15. ***(5 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 2.16. ***(5 т.)*** *Да се дефинира формално релацията $\stackrel{\beta}{\to}$ за Λ^* и да се докаже, че двете β-редущции са съгласувани, т.е. за произволен контекст от имена Γ*

(1) $\flat_\Gamma(M) \xrightarrow{\beta} \flat_\Gamma(N)$, *ако* $M, N \in \Lambda$, $\mathsf{FV}(MN) \subseteq \operatorname{dom}\Gamma$ *и* $M \xrightarrow{\beta} N$.
(2) $\#_\Gamma(M) \xrightarrow{\beta} P$, *ако* $M, N \in \Lambda_{|\Gamma|}$, $M \xrightarrow{\beta} N$ *и* $P \overset{\alpha}{=} \#_\Gamma(N)$.

Дефиниция 2.9 (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 2.17. ***(5 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 комбинаторна логика. Да се покаже, че за произволно $M \in \Lambda$:*

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

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

Дефиниция 2.10 (екстенсионално равенство)**.** Казваме, че M и N са екстенсионално равни и бележим $\lambda + \mathrm{ext} \models M = N$, ако

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

Задача 2.18. ***(3 т.)*** *Да се докаже, че релацията $\lambda + \mathrm{ext} \models M = N$ е λ-съвместима релация на еквивалентност.*

Задача 2.19. *Да се дефинират следните комбинатори и с помощта на индукция да се докаже формално тяхната коректност:*

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

Задача 2.20. ***(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 $c'_+ MN \overset{\beta}{\neq} c''_+ MN$.

Задача 2.21. ***(3 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 $C_I c_n \overset{\beta}{=} c_n$.*
(2) *Вярно ли е, че $C_I \overset{\beta\eta}{=} I$? Да се докаже или да се покаже контрапример.*

Задача 2.22. ***(3 т.)*** *Нека дефинираме*

$$\begin{array}{rcl} c_{\mathsf{tt}} & := & \lambda_{x,y}x \\ c_{\mathsf{ff}} & := & \lambda_{x,y}y \\ c_{\langle\rangle} & := & \lambda_{x,y,z}zxy \\ c_{\llcorner} & := & \lambda_p p c_{\mathsf{tt}} \\ c_{\lrcorner} & := & \lambda_p p c_{\mathsf{ff}} \\ c_P & := & \lambda_n c_{\lrcorner}\big(n(\lambda_z c_{\langle\rangle}(c_S(c_{\llcorner}z))(c_{\llcorner}z))(c_{\langle\rangle}c_0c_0)\big). \end{array}$$

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 2.23. ***(3 т.)*** *Нека дефинираме*

$$c_! := \lambda_n c_{\lrcorner}\big(n(\lambda_z c_{\langle\rangle}(c_S(c_{\llcorner}z))(c_*(c_S(c_{\llcorner}z))(c_{\lrcorner}z)))(c_{\langle\rangle}c_0c_1)\big).$$

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

Задача 2.24. ***(3 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 2.25. ***(3 т.)*** *Да се дефинират комбинатори* c_{quot} *и* c_{rem}, *така че за произволни* $m, n \in \mathbb{N}$:

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

Да се докаже формално коректността на дефинираните комбинатори.

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

Задача 2.26. ***(3 т.)*** *Да се дефинират комбинатори* $c_/$ *и* c_{prime}, *така че за произволни* $m, n \in \mathbb{N}$:

- $c_/ c_m c_n \stackrel{\beta}{=} c_{\exists_k(km=n)}$;
- $c_{\mathrm{prime}} c_n \stackrel{\beta}{=} c_{\neg\exists_{k,l>1}(kl=n)}$.

Да се докаже формално коректността на дефинираните комбинатори.

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

Задача 2.27. ***(2 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 2.28. ***(8 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 2.29. ***(3 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 2.30. ***(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 2.11 (λ-определимост)**.** Нека $f : \mathbb{N}^n \multimap \mathbb{N}$ 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 $Fc_{x_1} \dots c_{x_n}$ няма нормална форма.

Задача 2.31. ***(13 т.)*** *Да се докаже, че всяка частично рекурсивна функция е λ-определима.*

Задача 2.32. ***(21 т.)*** *Да се докаже, че всяка функция, изчислима с машина на Тюринг е λ-определима.*

Задача 2.33. ***(1 т.)*** *Да се покаже пример, че $\xrightarrow{\beta}$ не изпълнява свойството на диаманта.*

Задача 2.34. ***(2 т.)*** *Да се докаже, че ако $M \xrightarrow{\beta} M'$, то $M[x \mapsto N] \xrightarrow{\beta} M'[x \mapsto N]$*

Задача 2.35. ***(3 т.)*** *Да се докаже, че ако $N \xrightarrow{\beta} N'$, то $M[x \mapsto N] \stackrel{\beta}{\twoheadrightarrow} M[x \mapsto N']$*

Задача 2.36. ***(5 т.)*** *Да се покаже, че $\xrightarrow{\beta} \subseteq \xrightarrow{1} \subseteq \stackrel{\beta}{\twoheadrightarrow}$, т.е. че $M \xrightarrow{\beta} N$ влече $M \xrightarrow{1} N$ и че $M \xrightarrow{1} N$ влече $M \stackrel{\beta}{\twoheadrightarrow} N$.*

Задача 2.37. ***(3 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 2.38. ***(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).*

3. Типово λ-смятане

Дефиниция 3.1 (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 3.1. ***(3 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 3.2 (Изтриване на тип)**.** Дефинираме индуктивно изображение $|\cdot| : \Lambda^T \to \Lambda$, 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 3.2. ***(5 т.)*** *Да се покаже, че*

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

Дефиниция 3.3 (Слабо типизирани термове)**.** Дефинираме множеството на слабо типизираните 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 3.3. ***(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*
- $\Psi_\tau(\Gamma, \Phi_\tau(\Gamma, t^\tau)) = t^\tau$.

Задача 3.4. ***(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 3.5. ***(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 3.6. ***(8 т.)*** *Да се покаже, че η конверсията $M^{\rho\Rightarrow\sigma} \xrightarrow{\eta} \lambda_{x^\rho} M^{\rho\Rightarrow\sigma} x^\rho$ за $x \notin \mathsf{FV}(s)$ е конфлуентна и силно нормализируема. Да се обясни дали тези две свойства на η-конверсията са в сила за безтиповото λ-смятане и защо.*

Задача 3.7. ***(2 т.)*** *Да се покаже контрапример за това че от $M \stackrel{\beta}{=} N$ не следва, че $\Gamma \vdash M : \tau \iff \Gamma \vdash N : \tau$.*

Задача 3.8. ***(3 т.)*** *Да се докаже, че ако $\Gamma \vdash M : \tau$ и $M \xrightarrow{\beta} N$, то $\Gamma \vdash N : \tau$.*

Дефиниция 3.4 (Типова субституция)**.** Типова субституция наричаме всяко изображение $\xi : TV \to T$. 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 3.9. ***(2 т.)*** *Да се докаже, че ако $\vdash M : \tau$ и $\tau \supseteq \sigma$, то $\vdash M : \sigma$.*

Задача 3.10.

Задача 3.11. *2 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 3.12. ***(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 3.13. ***(3 т.)*** *Да се докаже, че ако $\Gamma \vdash M : \tau$ и $N \leq M$, тогава $\Gamma \vdash N : \sigma$ за някой тип σ.*

4. Теория на доказателствата

Задача 4.1. *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 4.2. ***(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 4.3. *Да се докажат законите на de Morgan в $G[123]c$ или Nc, а където е възможно в $G[123]m$ и Nm, 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: ***(1 т.)*** *Доказателствата да се опишат в λ-синтаксис или в системите* MINLOG *или* COQ.

Задача 4.4. *Да се докаже в $G[123]c$ или Nc, а където е възможно в $G[123]m$ или Nm, 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: ***(2 т.)*** *Доказателствата да се опишат в λ-синтаксис или в системите* MINLOG *или* COQ.

Задача 4.5. ***(3 т.)*** *Да се докажат Хилбертовите аксиоми на Hm в $G[123]m$ или Nm.*

Задача 4.6. ***(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:*

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

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 4.7. ***(3 т.)*** *Да се покаже, че за всяка формула A от аксиомите $\forall_{\vec{x}}(\perp \to P(\vec{x}))$ в Nm е изводима формулата $\perp \to A$.*

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

Задача 4.8. *Да се докаже в $G[123]c$ или Nc, че*

(1) ***(2 т.)*** $(A \to \exists_x B) \to \exists_x(A \to B)$, *ако* $x \notin \mathsf{FV}(A)$.
(2) ***(2 т.)*** $((A \to B) \to C) \to (A \to C) \to C$
(3) ***(2 т.)*** $(A \to B) \vee (B \to A)$

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

Задача 4.9. *Да се докаже, че:*

- ***(2 т.)*** $Ni \vdash (A \to B) \tilde{\vee} (A \to C) \to A \to B \vee C$
- ***(2 т.)*** $Nm \vdash (A \vee B \to C) \to (A \to C) \wedge (B \to C)$
- ***(2 т.)*** $Ni \vdash (A \to \tilde{\exists}_x B) \to \tilde{\exists}_x(A \to B)$, *ако* $x \notin \mathsf{FV}(A)$
- ***(2 т.)*** $Nm \vdash \exists_x(A \to B) \to \forall_x A \to B$, *ако* $x \notin \mathsf{FV}(B)$.

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

Задача 4.10. ***(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 4.11. *Да се построи доказателството формулата на пияниците:*

(1) ***(3 т.)*** $\exists_x(D(x) \to \forall_x D(x))$ *в* Nc *(силен вариант в класическа логика)*
(2) ***(3 т.)*** $\forall_x(\neg\neg D(x) \to D(x)) \to \tilde{\exists}_x(D(x) \to \forall_x D(x))$ *в* Nm *(слаб вариант в минимална логика).*

Екстра кредит: ***(1 т.)*** *Доказателствата да се опишат в* λ*-синтаксис или в системите* Minlog *или* Coq.

Задача 4.12. ***(5 т.)*** *Да се докаже, че* β*-редукцията в* $Nm(\to,\forall)$ *е локално конфлуентна, т.е. ако* $M_1 \overset{\beta}{\leftarrow} M \overset{\beta}{\to} M_2$*, то съществува доказателство* N*, за което* $M_1 \overset{\beta}{\twoheadrightarrow} N \overset{\beta}{\twoheadleftarrow} M_2$.