ЗАДАЧИ ЗА ИЗПИТ ПО λСТД (2018/19)

ТРИФОН ТРИФОНОВ

Правила за оценяване

- За успешно полагане на изпита са нужни ≥ 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. Структурна индукция

Задача 1.1. ***(2 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:*

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

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: 1 юли 2019 г.

Задача 1.3. ***(1 т.)*** *Нека $A \subseteq U$ е множество, дефинирана по индукция с клаузи $\{C_i\}$. Разглеждаме индуктивна дефиниция на подмножество F на $A \times U$, която следва клаузите C_i така, че за всеки генериран елемент в A се генерира точно един елемент в U. Да се покаже, че F е графика на тотална функция.*

Задача 1.4. ***(1 т.)*** *Разглеждаме оператора $\Gamma(X) := \{o\} \cup \{s(s(x)) \mid x \in X\}$. Да се покаже, че Γ е непрекъснат и да се намери най-малката неподвижна точка на Γ.*

Задача 1.5. *Да се дефинира оператор Γ, който е:*

- *строго растящ за крайни множества:* $\forall_{X-\text{ крайно}}\, X \subsetneq \Gamma(X)$
- *непрекъснат:* $\forall_{\{X_n\}_{n\in\mathbb{N}}} \bigcup_n \Gamma(X_n) = \Gamma(\bigcup_n X_n)$

и за който неподвижната точка X_Γ е:

(1) ***(2 т.)*** $\{x \mid x$ *е точен квадрат* $\}$
(2) ***(2 т.)*** $\{x \mid x$ *е точна степен на дадено числоа*$\}$
(3) ***(2 т.)*** $\{x \mid x$ *е просто число* $\}$
(4) ***(2 т.)*** $\{x \mid x$ *е число на Фибоначи* $\}$

Дефиниция 1.2. Дефинираме индуктивно множеството N:

- $o \in N$,
- ако $X \notin N$, то $s(X) \in N$.

Дефиниция 1.3. Дефинираме индуктивно събиране на елементи от N:

- $o + n := n$
- $s(n) + m := s(n + m)$

Задача 1.6. *Да се покаже, че:*

- ***(1 т.)*** $n + s(m) = s(n + m)$
- ***(1 т.)*** $m + n = n + m$

Дефиниция 1.4. Дефинираме индуктивно множеството L:

- $[] \in L$
- ако $n \in N, l \in L$, то $(n : l) \in L$.

Дефиниция 1.5. Дефинираме индуктивно дължина на списък от L:

- $len([]) := 0$
- $len(n : l) := 1 + len(l)$

Дефиниция 1.6. Дефинираме индуктивно конкатенация на списъци от L:

- $[] +\!\!+\ l := l$
- $(n : l_1) +\!\!+\ l_2 := n : (l_1 +\!\!+\ l_2)$

Задача 1.7. ***(1 т.)*** *Да се покаже, че* $\forall_{l_1,l_2}\, len(l_1 +\!\!+\ l_2) = len(l_1) + len(l_2)$

Задача 1.8. ***(2 т.)*** *Нека бинарната релация E се дефинира по индукция с клаузите $\{C_n\}$. Разглеждаме клаузите:*

(B) $\forall_{x,y}\, (x, y) \in E \to (x, y) \in R$
(R) $\forall_x\, (x, x) \in R$
(S) $\forall_{x,y}\, (x, y) \in R \to (y, x) \in R$
(T) $\forall_{x,y,z}\, (x, y) \in R \wedge (y, z) \in R \to (x, z) \in R$

Да се покаже, че релацията $E^{R,S,T}$, дефинирана чрез клаузите $\{B, R, S, T\}$, съвпада с релацията E', дефинирана чрез клаузите $\{C_n\} \cup \{R, S, T\}$.

2. Безтипово λ-смятане

2.1. Синтаксис на безтиповото λ-смятане.

Дефиниция 2.1 (Наивна субституция)**.** Нека $M, N \in \Lambda$, $x \in V$. Дефинираме субституцията на x с N в M, която ще отбелязваме с $M[x \rightsquigarrow N]$.

(1) $x[x \rightsquigarrow N] := N$
(2) $y[x \rightsquigarrow N] := y$ за $y \not\equiv x$
(3) $(M_1 M_2)[x \rightsquigarrow N] := (M_1[x \rightsquigarrow N])(M_2[x \rightsquigarrow N])$
(4) $(\lambda_x P)[x \rightsquigarrow N] := \lambda_x P$
(5) $(\lambda_y P)[x \rightsquigarrow N] := \lambda_y (P[x \rightsquigarrow N])$, ако $x \not\equiv y$

Дефиниция 2.2 (Частична субституция)**.** Дефинираме частичната субституция $M[x \hookrightarrow N]$ като в дефиниция 2.1 заместваме клаузата (5) с

(5) $(\lambda_y P)[x \hookrightarrow N] := \lambda_y (P[x \hookrightarrow N])$ за $y \not\equiv x$ и $x \notin \mathsf{FV}(P)$ или $y \notin \mathsf{FV}(N)$.

Задача 2.1. ***(2 т.)*** *Казваме, че наивната субституция $M[x \rightsquigarrow N]$ е* коректна, *ако $\mathsf{FV}(N) \cap \mathsf{BV}(M) = \emptyset$. Да се покаже, че:*

(1) *ако $M[x \rightsquigarrow N]$ е коректна, то $M[x \hookrightarrow N]$ е дефинирана $M[x \hookrightarrow N] \equiv M[x \rightsquigarrow N]$;*
(2) *има случай, в който $M[x \hookrightarrow N]$ е дефинирана, но $M[x \rightsquigarrow N]$ не е коректна;*

Дефиниция 2.3 (Субституция на Curry)**.** Дефинираме субституция на Curry $M[x \mapsto N]$ като в дефиниция 2.1 заместваме клаузата (5) с

(5) $(\lambda_y P)[x \mapsto N] := \lambda_z (P[y \mapsto z][x \mapsto N])$ във всички останали случаи, където $z \notin \mathsf{FV}(P) \cup \mathsf{FV}(N)$

Задача 2.2. ***(1 т.)*** *Клаузата (5) в субституцията на Curry нарушава шаблона на структурната индукция. Да се покаже, че въпреки това дефиницията е коректна, ако фиксираме избора на z.*

Задача 2.3. ***(2 т.)*** *Да се дефинира индуктивно релация на еквивалентност $\stackrel{\alpha}{=}$, така че $M \stackrel{\alpha}{=} N$ точно тогава, когато M може да се получи от N чрез подходящо преименуване на някои от свързаните си променливи.*

Задача 2.4. ***(1 т.)*** *Нека разгледаме фактор-множеството $\Lambda_{/\stackrel{\alpha}{=}}$, т.е. вместо отделни λ-термове разглеждаме класове на еквивалентност от λ-термове относно релацията $\stackrel{\alpha}{=}$. Да се докаже, че субституцията на Curry е функция над фактор-множеството $\Lambda_{/\stackrel{\alpha}{=}}$, т.е. ако $M \stackrel{\alpha}{=} M' \in \Lambda$, $N \stackrel{\alpha}{=} N' \in \Lambda$, то $M[x \mapsto N] \stackrel{\alpha}{=} M'[x \mapsto N']$.*

Задача 2.5. ***(1 т.)*** *Да се покаже, че винаги можем да намерим представител на класа на еквивалентност на даден терм, за който наивната субституция да е коректна, т.е. за всяко $M \in \Lambda$ можем да намерим $M' \stackrel{\alpha}{=} M$, така че $M'[x \rightsquigarrow N]$ да е коректна.*

Задача 2.6. ***(2 т.)*** *Да се покаже, че операцията за частична субституция може да се разглежда като тотална с точност до релацията $\stackrel{\alpha}{=}$, т.е.*

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

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

Задача 2.7. ***(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.*

2.2. **Безименни термове.**

Задача 2.8. ***(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 2.4 (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.

Задача 2.9. ***(1 т.)*** *Да се докаже, че за* $m < n$ *е вярно, че* $\Lambda_m \subsetneq \Lambda_n$.

Дефиниция 2.5. Нека $X \subseteq V$ е множество от променливи. Дефинираме $\Lambda_X := \{M \in \Lambda \mid \mathsf{FV}(M) \subseteq X\}$.

Задача 2.10. ***(3 т.)*** *Да се дефинират фамилиите от изображения* $\sharp_\Gamma : \Lambda_{|\Gamma|} \to \Lambda_{\mathrm{dom}\,\Gamma}$ *и* $\flat_\Gamma : \Lambda_{\mathrm{dom}\,\Gamma} \to \Lambda_{|\Gamma|}$ *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) $\sharp_\Gamma(\flat_\Gamma(M)) \stackrel{\alpha}{=} M$ *за всеки терм* $M \in \Lambda_{\mathrm{dom}\,\Gamma}$
(2) $\flat_\Gamma(\sharp_\Gamma(M)) \equiv M$ *за всеки терм* $M \in \Lambda_{|\Gamma|}$

Задача 2.11. ***(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 2.6 (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 2.7 (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 2.12. ***(3 т.)*** *Да се провери, че двете дефиниции за субституции са съгласувани относно изображенията $\sharp_\Gamma$ и $\flat_\Gamma$. За целта, нека фиксираме контекст от имена $\Gamma := x_{n-1}, \dots, x_0$. Да се покаже, че*

(1) $\sharp_\Gamma(M)[x_i \mapsto \sharp_\Gamma(N)] \stackrel{\alpha}{=} \sharp_\Gamma(M[i \mapsto N])$ *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* $M, N \in \Lambda_{\mathrm{dom}\,\Gamma}$.

Задача 2.13. ***(3 т.)*** *Да се направи програмна реализиация на субституция над безименни термове.*

Дефиниция 2.8 (Подтерм)**.** Дефинираме индуктивно функцията $Sub : \Lambda \Rightarrow 2^\Lambda$, където $Sub(M)$ е множеството на подтермовете на M индуктивно:

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

Дефинираме релацията "M е подтерм на N" по следния начин: $M \leq N := M \in Sub(N)$.

Задача 2.14. ***(3 т.)*** *Докажете, че релацията $\leq$ е частична наредба, т.е. че е рефлексивна, транзитивна и антисиметрична.* Упътване: покажете, че $M \leq N \iff Sub(M) \subseteq Sub(N)$.

Задача 2.15. ***(1 т.)*** *Да се направи програмна реализация на релацията за подтерм.*

Дефиниция 2.9 (Подтерм, алтернативна дефиниция №1)**.**

(1) $M \preceq M$
(2) Ако $MN \preceq P$, то $M \preceq P$ и $N \preceq P$
(3) Ако $\lambda_x M \preceq N$, то $M \preceq N$

Дефиниция 2.10 (Подтерм, алтернативна дефиниция №2)**.**

(1) $M \trianglelefteq M$
(2) Ако $M \trianglelefteq N$ и $P \in \Lambda$, то $M \trianglelefteq NP$ и $M \trianglelefteq PN$
(3) Ако $M \trianglelefteq N$ и $x \in V$, то $M \trianglelefteq \lambda_x N$

Задача 2.16. ***(3 т.)*** *Докажете, че трите дефиниции за подтерм са еквивалентни, т.е.* $M \leq N \iff M \preceq N \iff M \trianglelefteq N$.

Дефиниция 2.11 (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 2.17. ***(1 т.)*** *Дайте нетривиален (различен от $\Lambda^2, \equiv, \overset{\alpha}{=}$) пример за λ-съвместима релация.*

Задача 2.18. ***(1 т.)*** *Докажете, че R^λ е λ-съвместима за произволна релация R.*

Дефиниция 2.12 (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 2.19. *Да се докаже, че:*

(1) ***(1 т.)*** *$M \leq E[M]$ за произволни $M \in \Lambda$ и $E \in \Lambda^{[]}$*
(2) ***(1 т.)*** *Ако $M \leq N$, то съществува $E \in \Lambda^{[]}$ такова, че $E[M] \equiv N$*

Задача 2.20. ***(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 2.21. ***(3 т.)*** *Да се дефинира формално релацията $\overset{\beta}{\to}$ за множеството безименните термове Λ^* и да се докаже, че двете β-редукции са съгласувани, т.е. за произволен контекст от имена Γ*

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

Дефиниция 2.13 (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 2.22. ***(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) *$\mathsf{FV}(\Phi(M)) = \mathsf{FV}(M) \cup \{k, s\}$ 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 2.14 (екстенсионално равенство)**.** Казваме, че M и N са екстенсионално равни и бележим $\lambda + \mathrm{ext} \models M = N$, ако

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

Задача 2.23. ***(2 т.)*** *Да се докаже, че релацията* $\lambda + \mathrm{ext} \models M = N$ *е* λ*-съвместима релация на еквивалентност.*

2.3. **Изчисления в λ-смятането.**

Задача 2.24. *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* $c_{\mathrm{exp}} c_m c_n \stackrel{\beta}{=} c_{m^n}$ *за* $m, n \in \mathbb{N}$.
- ***(2 т.)*** c_{hyp}, *такъв че* $c_{\mathrm{hyp}} c_m c_n \stackrel{\beta}{=} c_p$, *където* $p = \underbrace{m^{m^{\cdot^{\cdot^{\cdot^{m}}}}}}_{n}$ *за* $m, n \in \mathbb{N}$.

Задача 2.25. ***(1 т.)*** *Нека са дадени следните дефиниции:*

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

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

Задача 2.26. ***(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 2.27. ***(2 т.)*** *Нека дефинираме*

$$\begin{array}{rcl} c_{\mathsf{tt}} & := & \lambda_{x,y} x \\ c_{\mathsf{ff}} & := & \lambda_{x,y} y \\ c_{\langle\rangle} & := & \lambda_{x,y,z} zxy \\ c_{\llcorner} & := & \lambda_p p c_{\mathsf{tt}} \\ c_{\lrcorner} & := & \lambda_p p c_{\mathsf{ff}} \\ c_P & := & \lambda_n c_{\lrcorner}\big(n(\lambda_z c_{\langle\rangle}(c_S(c_{\llcorner} z))(c_{\llcorner} z))(c_{\langle\rangle} c_0 c_0)\big). \end{array}$$

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 2.28. ***(2 т.)*** *Нека дефинираме*

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

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

Задача 2.29. ***(2 т.)*** *Да се дефинират комбинатори* $c_=$ *и* $c_<$, *за които за произволни* $m, n \in \mathbb{N}$:

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

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

Да се докаже формално коректността на дефинираните комбинатори.

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

Задача 2.30. ***(2 т.)*** *Да се дефинират комбинатори c_{quot} и c_{rem}, така че за произволни $m, n \in \mathbb{N}$:*

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

Да се докаже формално коректността на дефинираните комбинатори.

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

Задача 2.31. ***(3 т.)*** *Да се дефинират комбинатори $c_/$ и c_{prime}, така че за произволни $m, n \in \mathbb{N}$:*

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

Да се докаже формално коректността на дефинираните комбинатори.

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

Задача 2.32. ***(1 т.)*** *Да се дефинира комбинатор c_-, така че за произволни $m, n \in \mathbb{N}$ е изпълнено, че $c_- c_m c_n \overset{\beta}{=} c_{m \dot{-} n}$, 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 2.33. ***(5 т.)*** *Да се предложи дефиниция на списъци в безтиповото λ-смятане. С предложената дефиниция да се реализират:*

- *стандартните функции `length`, `append` и `member`*
- *функциите от по-висок ред `map`, `foldr` и `filter`.*

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

Задача 2.34. ***(3 т.)*** *Ако $f(\vec{x})$ и $g(\vec{x}, y, z)$ са λ-определими, да се покаже, че функцията h също е λ-определима:*

$$\text{(примитивна рекурсия)} \qquad \begin{aligned} h(\vec{x}, 0) &:= f(\vec{x}) \\ h(\vec{x}, y+1) &:= g(\vec{x}, y, h(\vec{x}, y)) \end{aligned}$$

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

Задача 2.35. ***(3 т.)*** *Да се дефинира комбинатор A, който реализира функцията на Акерман, т.е. за който*

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

Да се докаже формално, че предложеният комбинатор наистина реализира функцията на Акерман.

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

Задача 2.36. ***(1 т.)*** *Нека $\Theta := (\lambda_{x,F} F(xxF))(\lambda_{x,F} F(xxF))$. Да се провери, че $\Theta F \overset{\beta}{\twoheadrightarrow} F(\Theta F)$.*

Задача 2.37. ***(3 т.)*** *Да се дефинира комбинатор M, който симулира операцията "минимизация", т.е. ако t е комбинатор, за който съществува число n, такова че*

(1) $tc_n \overset{\beta}{=} c_0$

(2) $\forall_{m<n}\exists_k(tc_m \overset{\beta}{=} c_{k+1})$,

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

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

2.4. **Решими термове.**

Дефиниция 2.15 (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 $MN_1 \dots N_n \overset{\beta}{=} I$.

Дефиниция 2.16 (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 $Fc_{x_1} \dots c_{x_n} \overset{\beta}{=} c_y$;

(2) ако $f(x_1, \dots, x_n)$ не е дефинирана, то $Fc_{x_1} \dots c_{x_n}$ е нерешим.

Задача 2.38. ***(13 т.)*** *Да се докаже, че всяка функция, изчислима с машина на Тюринг е λ-определима.*

Екстра кредит: ***(8 т.)*** *Да се направи програмна реализация на превода от машина на Тюринг към λ-терм.*

Дефиниция 2.17 (Главна редукция)**.**

$\lambda_{\vec{x}}(\lambda_y P)Q\vec{N} \xrightarrow{h} \lambda_{\vec{x}}P[y \mapsto Q]\vec{N}$. $\overset{h}{\twoheadrightarrow} := (\xrightarrow{h})^{R,T}$.

Задача 2.39. ***(1 т.)*** *Да се направи програмна реализация на главна редукция.*

Задача 2.40. ***(2 т.)*** *Ако $M \xrightarrow{h} M'$, то $M[x \mapsto N] \xrightarrow{h} M'[x \mapsto N]$.*

Задача 2.41. ***(2 т.)*** *Ако $M \overset{h}{\not\to}$ и $M \overset{\beta}{\twoheadrightarrow} N$, то $N \overset{h}{\not\to}$.*

Задача 2.42. ***(1 т.)*** *Всеки терм има най-много една главна нормална форма.*

Дефиниция 2.18 (Индуктивна дефиниция на термове в нормална форма)**.** Дефинираме множеството $NF \subseteq \Lambda$:

(1) ако $x \in V, \vec{M} \in NF$, то $x\vec{M} \in NF$,

(2) ако $M \in NF$, то $\lambda_x M \in NF$.

Задача 2.43. ***(2 т.)*** $NF = \left\{M \in \Lambda \mid M \overset{\beta}{\not\to}\right\}$.

2.5. **Нормализация и конфлуентност.**

Задача 2.44. ***(1 т.)*** *Да се покаже пример, че $\xrightarrow{\beta}$ не изпълнява свойството на диаманта.*

Задача 2.45. ***(1 т.)*** *Да се докаже, че ако графът на редукциите на терма M е краен и ацикличен, то M е силно нормализируем.*

Задача 2.46. ***(2 т.)*** *Нека $D \subseteq \Lambda^2$ е произволна редукция. Тогава ако $(N, N') \in D^{\lambda,R,T}$, то $(M[x \mapsto N], M[x \mapsto N']) \in D^{\lambda,R,T}$ за произволен терм $M \in \Lambda$.*

Задача 2.47. ***(2 т.)*** *Да се докаже, че ако* $M \xrightarrow{\beta} M'$*, то* $M[x \mapsto N] \xrightarrow{\beta} M'[x \mapsto N]$

Дефиниция 2.19. (1) $M \xrightarrow{1} M$ за всяко $M \in \Lambda$

(2) ако $M \xrightarrow{1} M'$, то $\lambda_x M \xrightarrow{1} \lambda_x M'$

(3) ако $M \xrightarrow{1} M'$ и $N \xrightarrow{1} N'$, то $MN \xrightarrow{1} M'N'$

(4) ако $M \xrightarrow{1} M'$ и $N \xrightarrow{1} N'$, то $(\lambda_x M)N \xrightarrow{1} M'[x \mapsto N']$

Задача 2.48. ***(3 т.)*** *Да се покаже, че* $\xrightarrow{\beta} \subseteq \xrightarrow{1} \subseteq \overset{\beta}{\twoheadrightarrow}$*, т.е. че* $M \xrightarrow{\beta} N$ *влече* $M \xrightarrow{1} N$ *и че* $M \xrightarrow{1} N$ *влече* $M \overset{\beta}{\twoheadrightarrow} N$*.*

Задача 2.49. ***(2 т.)*** *Ако* $M \overset{\eta}{\twoheadrightarrow} M'$*, то* $M[x \mapsto N] \overset{\eta}{\twoheadrightarrow} M'[x \mapsto N]$*.*

Задача 2.50. ***(5 т.)*** $\xrightarrow{\eta}$ *е конфлуентна.*

Задача 2.51. ***(5 т.)*** $\overset{\beta}{\twoheadrightarrow}$ *комутира с* $\overset{\eta}{\twoheadrightarrow}$*.* Упътване: Достатъчно е да видим, че $\xrightarrow{\beta}$ комутира с $\xrightarrow{\eta}$.

2.6. **Стратегии за редукция.**

Дефиниция 2.20 (Стратегия за редукция)**.** Нека $\Lambda^\perp := \Lambda \cup \{\perp\}$, където $\perp \notin \Lambda$. Нека $\Phi : \Lambda \to \Lambda^\perp$ е такава, че:

- ако $\Phi(M) \not\equiv \perp$, то $M \xrightarrow{\beta} \Phi(M)$,
- ако $\Phi(M) \equiv \perp$, то $M \not\xrightarrow{\beta}$,

тогава Φ наричаме *стратегия за редукция.*
Дефинираме частичната функция $\Phi^*(M) : \Lambda \rightharpoonup \Lambda$

$$\Phi^*(M) := \begin{cases} \Phi^*(\Phi(M)), & \text{ако } \Phi(M) \not\equiv \perp, \\ M, & \text{ако } \Phi(M) \equiv \perp \end{cases}$$

Дефиниция 2.21 (Нормална стратегия)**.** Дефинираме

$$\mathsf{NR}(M) := \begin{cases} \perp, & \text{ако } M \not\xrightarrow{\beta}, \\ \lambda_x \mathsf{NR}(N), & \text{ако } M \equiv \lambda_x N, \mathsf{NR}(N) \not\equiv \perp \\ P[x \mapsto Q], & \text{ако } M \equiv (\lambda_x P)Q, \\ \big(\mathsf{NR}(P)\big)Q, & \text{ако } M \equiv PQ, P \not\equiv \lambda_x P', \mathsf{NR}(P) \not\equiv \perp, \\ P\big(\mathsf{NR}(Q)\big), & \text{ако } M \equiv PQ, \mathsf{NR}(P) \equiv \perp, \mathsf{NR}(Q) \not\equiv \perp. \end{cases}$$

Задача 2.52. ***(1 т.)*** *Да се докаже, че* $\mathsf{NR}(\cdot)$ *е стратегия за редукция.*

Задача 2.53. ***(3 т.)*** *Да се направи програмна реализация на* $\mathsf{NR}(\cdot)$*.*

Дефиниция 2.22 (Апликативна стратегия)**.** Дефинираме

$$\mathsf{AR}(M) := \begin{cases} \perp, & \text{ако } M \not\xrightarrow{\beta}, \\ \lambda_x \mathsf{AR}(N), & \text{ако } M \equiv \lambda_x N, \mathsf{AR}(N) \not\equiv \perp \\ \big(\mathsf{AR}(P)\big)Q, & \text{ако } M \equiv PQ, \mathsf{AR}(P) \not\equiv \perp, \\ P\big(\mathsf{AR}(Q)\big), & \text{ако } M \equiv PQ, \mathsf{AR}(P) \equiv \perp, \mathsf{AR}(Q) \not\equiv \perp, \\ P[x \mapsto Q], & \text{ако } M \equiv (\lambda_x P)Q, \mathsf{AR}(P) \equiv \mathsf{AR}(Q) \equiv \perp. \end{cases}$$

Задача 2.54. ***(1 т.)*** *Да се докаже, че* $\mathsf{AR}(\cdot)$ *е стратегия за редукция.*

Задача 2.55. ***(3 т.)*** *Да се направи програмна реализация на* $\mathsf{AR}(\cdot)$*.*

3. Типово λ-смятане

3.1. Синтаксис на типовото λ-смятане.

Задача 3.1. ***(1 т.)*** *Докажете, че $\omega \equiv \lambda_x xx$ и $\Omega \equiv \omega\omega$ не са типизируеми.*

Задача 3.2. ***(2 т.)*** *Ако $\Gamma \vdash M : \tau$, то $\mathsf{FV}(M) \subseteq \operatorname{dom}\Gamma$.*

Задача 3.3. ***(2 т.)*** *Нека $\Gamma \restriction X := \{x : \tau \in \Gamma \mid x \in X\}$. Ако $\Gamma \vdash M : \tau$, то $\Gamma \restriction \mathsf{FV}(M) \vdash M : \tau$.*

Задача 3.4. ***(2 т.)*** *Ако $\Gamma \vdash M : \tau$ и $N \leq M$, то $\exists_{\Delta,\sigma}$ такива, че $\Delta \vdash N : \sigma$.*

Задача 3.5. ***(3 т.)*** *$\Gamma \vdash M : \tau$ тогава и само тогава, когато съществува типов извод с корен $M : \tau$, чиито незадраскани листа са в Γ и съдържат само променливи от $\mathsf{FV}(M)$.*

Задача 3.6. ***(2 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 3.7. *Да се докаже, че:*

- ***(2 т.)*** *ако $\Gamma \vdash M : \tau$ и $M \xrightarrow{\beta} N$, то $\Gamma \vdash N : \tau$*
- ***(2 т.)*** *ако $\Gamma \vdash M : \tau$ и $M \xrightarrow{\eta} N$, то $\Gamma \vdash N : \tau$*

Дефиниция 3.1 (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 3.8. ***(2 т.)*** *Да се докаже, че ако $\vdash M : \tau$ и $\tau \supseteq \sigma$, то $\vdash M : \sigma$.*

Задача 3.9. ***(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 3.10. ***(2 т.)*** *Да се докаже, че ако $\Gamma \vdash M : \tau$ и $N \leq M$, тогава $\Gamma \vdash N : \sigma$ за някой тип σ.*

Задача 3.11. ***(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.*

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 3.12. *Да се покаже, че*

(1) ***(2 т.)*** *за всеки затворен типизиран терм $M^\tau \in \Lambda^T$ може да се намери типов извод на типовото съждение $\|M^\tau\| : \tau$.*

(2) ***(2 т.)*** *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 $\|N\| \equiv M$.*

Екстра кредит: ***(5 т.)*** *Да се реализират типизиране λ-термове в стил Church и да се реализира конвертирането на нетипизирани в типизиране λ-термове и обратно.*

Задача 3.13. ***(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*
- *$\Psi_\tau(\Gamma, \Phi_\tau(\Gamma, t^\tau)) = t^\tau$.*

Екстра кредит: ***(5 т.)*** *Да се направи програмна реализация на безименни типизирани λ-термове.*

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 3.14. ***(2 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 3.15. ***(5 т.)*** *Да се формулира и докаже еквивалентността между Λ^{WT} и Λ^T.*

Задача 3.16. ***(5 т.)*** *Да се направи програмна реализация на слабо типизирани λ-термове.*

Задача 3.17. ***(2 т.)*** *Да се докаже, че типовите променлива α не са обитаеми.* Упътване: използвайте теоремата за силната нормализация.

3.2. **Силна нормализация.**

Дефиниция 3.4 (силно изчислими термове)**.** Дефинираме SC^τ („силно изчислим в типа τ") с индукция по τ:

- $\mathsf{SC}^\alpha := \mathsf{SN}^\alpha$,
- $\mathsf{SC}^{\rho\Rightarrow\sigma} := \{M^{\rho\Rightarrow\sigma} \mid \forall_{N^\rho}(N^\rho \in \mathsf{SC}^\rho \to (MN)^\sigma \in \mathsf{SC}^\sigma)\}$.

$$\mathsf{SC} := \bigcup_{\tau \in T} \mathsf{SC}^\tau.$$

Задача 3.18. ***(1 т.)*** *$M \in \mathsf{SC}^{\vec{\rho}\Rightarrow\sigma}$ точно тогава, когато $\forall_{N_i \in \mathsf{SC}^{\rho_i}}\big(M\vec{N} \in SC^{\sigma}\big)$.*

Задача 3.19. ***(1 т.)*** *Ако $M, N, \vec{P} \in \mathsf{SN}$ и $M[x \mapsto N]\vec{P} \in \mathsf{SC}^{\tau}$, то $(\lambda_x M)N\vec{P} \in \mathsf{SC}^{\tau}$.*

Дефиниция 3.5 (Ниво на тип)**.** За произволен тип $\tau \in T$ дефинираме индуктивно *ниво на типа* τ, което бележим с $lvl(\tau)$:

- $lvl(\alpha) := 0$
- $lvl(\rho \Rightarrow \sigma) := \max\big(lvl(\rho) + 1, lvl(\sigma)\big)$.

Задача 3.20. ***(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.*

3.3. **Типови системи.**

Задача 3.21. ***(2 т.)*** *Да се реализират всички операции за сравнение на естествени числа в системите T или PCF.*

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

Задача 3.22. ***(3 т.)*** *Да се реализират операциите за частно и остатък от целочислено деление в системите T или PCF.*

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

Задача 3.23. ***(3 т.)*** *Да се реализират предикатите за проверка за делимост и проверка за простота на число в системите T или PCF.*

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

Задача 3.24. ***(3 т.)*** *Да се реализира функцията на Акерман в системата T.*

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

Задача 3.25. ***(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.*

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

3.4. **Нормализация чрез оценяване.**

Дефиниция 3.6 (λ-оценка)**.** Нека е дадена λ-интерпретация на типовете. Фамилията от функции $\xi := \{\xi_\tau\}_{\tau\in\overline{T}}$, за които $\xi_\tau : V^\tau \to [\![\tau]\!]$ наричаме *λ-оценка.*

Дефиниция 3.7 (стойност при оценка)**.** Нека ξ е λ-оценка в теоретико-множествената интерпретация на типовете. Дефинираме индуктивно $[\![M^\tau]\!]_\xi \in [\![\tau]\!]$ (*стойност на терма M^τ при оценка ξ*):

- $[\![x]\!]_\xi := \xi(x)$,
- $[\![M_1 M_2]\!]_\xi := [\![M_1]\!]_\xi([\![M_2]\!]_\xi)$,
- $[\![\lambda_{x^\rho} N^\sigma]\!]_\xi := f : [\![\rho]\!] \to [\![\sigma]\!]$, където $f(a) := [\![N]\!]_{\xi^a_x}$.

Задача 3.26. ***(2 т.)*** *Покажете, че ако $\xi(x) = \nu(x)$ за всяко $x \in \mathsf{FV}(M)$, то $[\![M]\!]_\xi = [\![M]\!]_\nu$.*

Задача 3.27. ***(5 т.)*** *Покажете, че теоретико-множествената интерпретация е λ-модел.*

Упътване: Индукция по $\overset{\beta\eta}{=}$, с доказване на помощна Лема за субституцията:

$$[\![M[x \mapsto N]]\!]_\xi = [\![M]\!]_{\xi_x^{[\![N]\!]_\xi}}.$$

Дефиниция 3.8 (η-дълга нормална форма на терм)**.** Нека $M^\tau \in NF$ е терм в β-нормална форма. Дефинираме $lnf(M^\tau)$ с едновременна индукция по $M \in NF$ и τ:

- ако $M \equiv x\vec{N}$, нека $N_i' := lnf(N_i)$:
  - ако $\tau \equiv \alpha$, то $lnf(x\vec{N}) := x\vec{N'}$,
  - ако $\tau \equiv \rho \Rightarrow \sigma$, то $lnf(x\vec{N}) := \lambda_{y^\rho} lnf\big(x\vec{N'}\, lnf(y)\big)$ за свежа $y^\rho \in V^T$,
- ако $M \equiv \lambda_x N$, то $lnf(\lambda_x N) := \lambda_x lnf(N)$.

Задача 3.28. ***(3 т.)*** *Покажете, че $lnf : NF \to NF$, т.е. lnf запазва β-нормалната форма.*

Дефиниция 3.9 (Индуктивна дефиниция на термове в дълга нормална форма)**.** Дефинираме множеството $LNF \subseteq \Lambda^T$

(1) ако $x \in V^T, \vec{M} \in LNF$ и $(x\vec{M})^\alpha$, то $x\vec{M} \in LNF$,
(2) ако $M \in LNF$, то $\lambda_x M \in LNF$.

Задача 3.29. ***(3 т.)*** *$LNF = \big\{M \in \Lambda^T \mid lnf(M) \equiv M\big\}$.*

Дефиниция 3.10 (Отражение и реификация)**.** Едновременно дефинираме с индукция по типа τ:

- $\Uparrow_\mu (M) := M$,
- <.-> $\Uparrow_{\rho\Rightarrow\sigma} (M)(a) := \Uparrow_\sigma (M \Downarrow_\rho (a))$,
- $\Downarrow_\mu (M) := M$,
- <.-> $\Downarrow_{\rho\Rightarrow\sigma} (a) := \lambda_{x^\rho} \Downarrow_\sigma (a(\Uparrow_\rho x))$, където x^ρ е свежа променлива.

Задача 3.30. ***(1 т.)*** *Да се докаже, че*

$$\Uparrow_{\vec{\rho}\Rightarrow\sigma} (M)(a_1)(a_2)\dots(a_n) := \Uparrow_\sigma (M \Downarrow_{\rho_1} (a_1) \Downarrow_{\rho_2} (a_2) \dots \Downarrow_{\rho_n} (a_n))$$

Задача 3.31. ***(3 т.)*** *Да се дефинира $lnf : \Lambda^{T*} \to NF^*$ за безименни термове.*

Задача 3.32. ***(8 т.)*** *Да се направи програмна реализация на алгоритъма за нормализация чрез оценяване на безтипово λ-смятане.*

4. Теория на доказателствата

4.1. **Системи за изразяване на доказателства.**

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 (теорема за генерализацията)**.** ***(3 т.)*** *Ако $x \notin \mathsf{FV}[\Gamma]$, то $\Gamma \vdash \forall_x A$ тогава и само тогава когато $\Gamma \vdash A$.*

Задача 4.3. ***(3 т.)*** *Да се докажат Хилбертовите аксиоми на Hm в $G[123]m$ или Nm.*

Задача 4.4. ***(2 т.)*** *Като се използва предишната задача, да се докаже че $\Gamma \vdash_{H[mic]} A \Rightarrow \Gamma \vdash_{G1[mic]} A$.*

Задача 4.5. ***(2 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 4.6. *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: ***(по 1 т.)*** *Доказателствата да се опишат в λ-синтаксис или в системите* Minlog *или* Coq.

Задача 4.7. ***(2 т.)*** *Да се докаже, че:* $\vdash_i (\neg\neg A \to \neg\neg B) \to \neg\neg(A \to B)$.

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.)*** $(A \to B) \vee (B \to A)$
(4) ***(1 т.)*** $((A \to B) \to A) \to A$ *(закон на Peirce)*
(5) ***(2 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 на формулата за пияниците)*

Екстра кредит: ***(по 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.)*** *Да се покажат всички пътеки в доказателството и да се нормализира, ако не е нормално.*

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

Задача 4.10. ***(5 т.)*** *Да се докаже, че ако съществува извод на $\Gamma \Rightarrow \Delta$ в G2[mic] с дълбочина n, то за произволни Γ' и Δ' съществува извод на $\Gamma\Gamma' \Rightarrow \Delta\Delta'$ в G2[mic] с дълбочина n.*

Задача 4.11. ***(5 т.)*** *Да се покаже, че* $\vdash_{G1[mic]} \Gamma \Rightarrow \Delta \iff \vdash_{G2[mic]} \Gamma \Rightarrow \Delta$.
Упътване: ($\Leftarrow$) е тривиална; за ($\Rightarrow$) елиминираме последователно прилаганията на LW и RW отгоре надолу използвайки предишното твърдение.

Дефиниция 4.1 (Обратимо правило)**.** Правилото $\dfrac{S_1}{S_2}$ наричаме *обратимо*, ако $\vdash S_1 \iff \vdash S_2$.

Задача 4.12 (Обратимост в G3). ***(5 т.)*** *Да се покаже, че всички правила в G3[mic] са обратими, с изключение на $L\to$ в G3[mi], за което само десният клон е обратим.*

Задача 4.13 (Симулация на контракция). ***(8 т.)*** *Да се покаже, че LC е изводимо в G3[mic] и RC е изводимо в G3c.*

Задача 4.14. ***(8 т.)*** *Да се напишат доказателства в HA^ω, че операциите от задача 3.21 за сравнение са съответно строга/частична наредба (за $<$, $>$, $\leq$, $\geq$) и релация на еквиваленност (за $=$).*

Екстра кредит: ***(2 т.)*** *Доказателствата да се опишат в λ-синтаксис или в системите* Minlog *или* Coq.

Задача 4.15. ***(5 т.)*** *Да се напишат доказателства в HA^ω, които доказват коректността на операциите от задача 3.22 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 4.16. ***(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.

Екстра кредит: ***(5 т.)*** *С помощта на интерпретацията за реализуемост да се извлече изчислителния смисъл на доказателството.*

4.2. **Влагане на класическа в минимална логика.**

Задача 4.17. *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: ***(по 1 т.)*** *Доказателствата да се опишат в λ-синтаксис или в системите* Minlog *или* Coq.

Задача 4.18. ***(3 т.)*** *Да се покаже, че за всяка формула A от аксиомите $\mathsf{efq}_p \forall_{\vec{x}}(\bot \to p\vec{x})$ в Nm е изводима формулата $\bot \to A$.*

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 4.19. ***(3 т.)*** *Да се докаже, че $\mathrm{HA}^\omega \vdash_m F \to A$.*

Задача 4.20. ***(3 т.)*** *Да се докаже, че $\mathrm{HA}^\omega \vdash_m^{\to\forall} ((A \to F) \to F) \to A$.*

Задача 4.21. ***(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 4.22. ***(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* $\Gamma^q \vdash_m A^q$

4.3. **Нормални доказателства.**

Задача 4.23. ***(2 т.)*** *Да се докаже, че всяка формула в дадено нормално доказателство M в $Nm(\to\forall)$ принадлежи на някоя пътека.*

Дефиниция 4.2 (Подформула)**.** За две формули A и B дефинираме индуктивно релацията "A е подформула на B"' (бележим $A \leq B$):

- $A \leq A$.
- Ако $A \wedge B \leq C$, $A \vee B \leq C$ или $A \to B \leq C$, то $A \leq C$ и $B \leq C$.
- Ако $\forall_x A \leq B$ или $\exists_x A \leq B$, а t е произволен терм, то $A[x \mapsto t] \leq B$.

Задача 4.24. ***(5 т.)*** *Нека е дадено нормално доказателство M в $Nm(\to\forall)$ на формулата C от допусканията $A_i^{u_i}$. С индукция по пътеките да се докаже, че то всяко срещане на формула в доказателството е подформула на някое от C или A_i.*

Задача 4.25. ***(2 т.)*** *Да се докаже, че във всяка пътека $\pi = A_0, \ldots, A_n$ в $Nm(\to\forall)$ съществува минимална формула A_i, така че*

- $\forall_{0 \leq j < i}(A_{j+1} \leq A_j)$ *и правилото с предпоставка A_j и заключение A_{j+1} е правило за елиминация ("$-$" правило).*
- $\forall_{i < j \leq n}(A_{j-1} \leq A_j)$ *и правилото с предпоставка A_{j-1} и заключение A_j е правило за въвеждане ("$+$" правило).*

Задача 4.26. ***(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 $M_1 \overset{\beta}{\twoheadrightarrow} N \overset{\beta}{\twoheadleftarrow} M_2$.*

Задача 4.27. ***(13 т.)*** *Да се реализира програма, която нормализира дадено доказателство в $Nm(\to\forall)$ чрез оценяване (NbE).*

Задача 4.28. ***(5 т.)*** *Да се разпишат всички 15 пермутиращи редукции за правилата, породени от аксиомите $\wedge^-$, $\vee^-$, $\exists^-$ над правилата, породени от аксиомите $\to^-$, $\forall^-$, $\wedge^-$, $\vee^-$, $\exists^-$ в двата записа: дърво на извод и термов синтаксис.*

Задача 4.29. ***(3 т.)*** *Да се докаже, че всяка формула в дадено нормално доказателство M в $Nm(\to\wedge\vee\forall\exists)$ принадлежи на някоя пътека.*

Задача 4.30. ***(3 т.)*** *В нормално доказателство съществува минимален сегмент, който служи за разделителна линия между последователни прилагания на $\circ^-$ и $\circ^+$ правила.*