

Задачи по изчислимост и сложност

Йоан Василев

2 февруари 2026 г.

1 Задачи за ЕК

Задача 1 (*nested recursion*). Казваме, че функцията h се определя с вложена рекурсия (nested recursion) от функциите f, g, π , ако за нея е изпълнено:

$$\left| \begin{array}{l} h(0, y) \simeq f(y) \\ h(x+1, y) \simeq g(x, y, h(x, \pi(x, y))) \end{array} \right.$$

Докажете, че ако f, g, π са примитивно рекурсивни, то h също е.

Решение. Дефинираме функция $H(x, y, z)$ по следния начин:

$$H(x, y, z) \simeq \begin{cases} h(x, y), & \text{ако } z = 0 \\ h(x, \pi(x, \pi(x+1, \dots, \pi(x+z-1, y) \dots)), & \text{иначе} \end{cases}$$

Ще докажем, че H е примитивно рекурсивна.

- При $x = 0$ имаме:

$$H(0, y, 0) = \begin{cases} h(0, y) = f(y), & \text{ако } z = 0 \\ h(0, \pi(0, \pi(1, \dots, \pi(z-1, y) \dots)) = f(\pi(0, \pi(1, \dots, \pi(z-1, y) \dots)), & \text{иначе} \end{cases}$$

Горното ще служи за базов случай на примитивната рекурсия за H , като за целта трябва да докажем, че функцията на два аргумента $\Phi(y, z) := H(0, y, z)$ е примитивно рекурсивна. Това се вижда лесно от $\Phi(y, z) = f(F(z, y, z))$ за F дефинирана по примитивно рекурсивната схема:

$$\left| \begin{array}{l} F(0, y, z) = y \\ F(n+1, y, z) = \pi(z-n, F(n, y, z)) \end{array} \right.$$

- Отделно имаме и следната зависимост:

$$\begin{aligned} H(x+1, y, z) &\simeq h(x+1, \underbrace{\pi(x+1, \pi(x+2, \dots, \pi(x+z, y) \dots))}_t) \simeq g(x, y, h(x, \pi(x, t))) = \\ &= g(x, y, \underbrace{h(x, \pi(x, \pi(x+1, \pi(x+2, \dots, \pi(x+z, y) \dots))))}_{H(z,y,z+1)}) \simeq g(x, y, H(x, y, z+1)) \end{aligned}$$

От направеното досега получихме:

$$\left| \begin{array}{l} H(0, y, z) = \Phi(y, z) \\ H(x+1, y, z) = g(x, y, H(x, y, z+1)) \end{array} \right.$$

Ще кодираме x, z в един аргумент, за да бъде рекурсията само по него. Нека $\langle \Pi, L, R \rangle$ е кодираща схема ($\Pi(x, z) = 3^x 2^z$ ще ни свърши работа) и $H(x, y, z) = \tilde{H}(\Pi(x, z), y)$, по-точно $\tilde{H}(t, y) = H(L(t), y, R(t))$ за $t = 2^k 3^l$ и $H(t, y) = 0$ в противен случай. Тогава горната система, записана за \tilde{H} , придобива вид:

$$\left| \begin{array}{l} \tilde{H}(\underbrace{\Pi(0, z)}_{2^z}, y) = \Phi(y, R(\Pi(0, z))) \\ \tilde{H}(\Pi(x+1, z), y) = g(L(\Pi(x+1, z)-1, y, \tilde{H}(\Pi(L(\Pi(x+1, z))-1, R(\Pi(x+1, z))+1)), y) \end{array} \right.$$

Опростено, това изглежда по следния начин:

$$\tilde{H}(t, y) = \begin{cases} \Phi(y, R(t)), & \text{ако } t = 2^k 3^l \& l = 0 \\ g(L(t) - 1, y, \underbrace{\tilde{H}(\Pi(L(t) - 1, R(t) + 1), y)}_{G(t)}), & \text{ако } t = 2^k 3^l \& l \neq 0 \\ 0, & \text{иначе.} \end{cases}$$

Участващите функции и предикати са примитивно рекурсивни. Ключовото наблюдение е, че $G(t) < t$ /понеже $\Pi(L(t) - 1, R(t) + 1) < t = \Pi(L(t), R(t))$, т.e. $\tilde{H}(t, y)$ е примитивно рекурсивна по схемата за силна рекурсия (по-точно следствие от нея), откъдето $H(x, y, z) = \tilde{H}(\Pi(x, z), y)$ също е такава. Върно е, че $h(x, y) = H(x, y, 0)$, следователно и $h(x, y)$ е примитивно рекурсивна. ■

Задача 2 (*Непримитивност на функцията на Акерман*). Докажете, че функцията на Акерман не е примитивно рекурсивна.

$$\left| \begin{array}{l} F(0, y) = y + 1 \\ F(x + 1, 0) = F(x, 1) \\ F(x + 1, y + 1) = F(x, F(x + 1, y)) \end{array} \right.$$

Решение. Да започнем с доказване на няколко помощни твърдения за функцията на Акерман.

Лема 1

Функцията на Акерман е монотонно растяща по двата си аргумента.

Доказателство. Може да се докаже едновременно (и за двата аргумента) с индукция по x и вложена в нея по y .

За $x = 0$ нещата са тривиални, $F(0, y) = y + 1 < y + 2 = F(0, y + 1)$.

При $x > 0, y = 0 : F(x, 0) = F(x - 1, 1) \xrightarrow{\text{инд. хипотеза}} F(x - 1, 0)$.

За $x \neq 0, y \neq 0 :$

$$F(x, y) = F(x - 1, F(x, y - 1)) \xrightarrow{\text{инд. хипотеза}} F(0, F(x, y - 1)) = F(x, y - 1) + 1 > F(x, y - 1).$$

$$\text{Също така и } F(x - 1, F(x, y - 1)) \geq F(x - 1, \underbrace{F(1, y - 1)}_{> F(0, y - 1) = y}) \xrightarrow{\text{инд. хипотеза}} F(x - 1, y). \quad \square$$

Лема 2

За всеки x, y е сила $F(x + 1, y) > y + 1$.

Доказателство. Ползвайки монотонността, получаваме $F(x + 1, y) > F(x, y) \geq F(0, y) = y + 1$. □

Лема 3

За всеки x, y е сила $F(x, y + 1) \leq F(x + 1, y)$.

Доказателство. Доказваме с индукция едновременно по x (външно) и y (вътрешно). За $x = 0$ твърдението следва от $F(0, y + 1) = y + 2 \stackrel{\text{Лема 2}}{\leq} F(1, y)$, а при $y = 0 : F(x, y + 1) = F(x, 1) = F(x + 1, 0)$. В останалите случаи, за $x, y \geq 1$:

$$F(x, y + 1) = F(x - 1, F(x, y)) \stackrel{\text{МОНОТОННОСТ}}{\leq} F(x, F(x, y)) \stackrel{\text{ИП, МОНОТОН.}}{\leq} F(x, F(x + 1, y - 1)) = F(x + 1, y)$$

□

Лема 4

За всяко x съществува z такова, че за всяко y : $F(x, 2y) \leq F(z, y)$.

Доказателство. Твърдим, че $z = x + 2$ удовлетворява исканото. Доказваме с вложена индукция първо по x , после по y .

За $y = 0$ твърдението следва директно, $F(x, 2 \cdot 0) = F(x, 0) \leq F(x + 2, 0)$.

В останалите случаи, за $y \geq 1$:

$$\begin{aligned} F(x + 2, y) &= F(x + 1, F(x + 2, y - 1)) \stackrel{\text{инд. хипотеза}}{\geq} F(x + 1, \underbrace{F(x, 2y - 2)}_{\geq 2y-1}) \geq \\ &\stackrel{\text{Лема 3}}{\geq} F(x, 2y) \end{aligned}$$

□

Обратно към същинското доказателство. Ще докажем по индукция, че за всяка примитивно функция g е в сила $P(g) \Rightarrow$ съществува естествено k такова, че за всяко $\bar{x} \in \mathbb{N}^n$:

$$g(x_1, \dots, x_n) < F(k, \max\{x_1, \dots, x_n\})$$

База:

- $\mathcal{O}(x)$: при $k = 0$ за всяко x имаме $\mathcal{O}(x) = 0 < x + 1 = F(0, x) = F(k, \max\{x\})$; ✓
- $S(x)$: От Лема 2 имаме, че $S(x) = x + 1 < F(1, x)$ за всяко x ; ✓
- $I_k^n(\bar{x})$: $I_k^n(\bar{x}) = x_k \leq \underbrace{\max\{x_1, \dots, x_n\}}_m < m + 1 = F(0, m)$, т.e. при $k = 0$ неравенството е в сила; ✓

ИС:

- **примитивна рекурсия:** Нека $g \in \mathcal{F}_n, f \in \mathcal{F}_{n+2}$ са примитивно рекурсивни, като $P(f)$ и $P(g)$, а $h \in \mathcal{F}_{n+1}$ е получена с примитивна рекурсия от f и g , т.e.

$$\left| \begin{array}{l} h(\bar{x}, 0) = g(\bar{x}) \\ h(\bar{x}, y + 1) = f(\bar{x}, y, h(\bar{x}, y)) \end{array} \right.$$

По предположение са в сила следните:

$$P(g) \Rightarrow \exists k_1 \forall \bar{x}: g(\bar{x}) < F(k_1, \max\{x_1, \dots, x_n\})$$

$$P(f) \Rightarrow \exists k_2 \forall \bar{x}, y, z: f(\bar{x}, y, z) < F(k_2, \max\{x_1, \dots, x_n, y, z\})$$

Тук ще ползваме, че функцията на Акерман е растяща по двата си аргумента, така че горните неравенства остават в сила и за $k := \max\{k_1, k_2\}$. Първо ще покажем, че $h(\bar{x}, y) < F(k + 1, \max\{x_1, \dots, x_n, y\} + y)$ за всеки \bar{x}, y с индукция по y :

База: При $y = 0$ за всяко \bar{x}

$$h(\bar{x}, 0) = g(\bar{x}) < F(k, \max\{x_1, \dots, x_n, 0\}) \leq F(k + 1, \max\{x_1, \dots, x_n, 0\} + 0)$$

ИП: За всяко $z < y$ и всяко \bar{x} : $h(\bar{x}, z) < F(k + 1, \max\{x_1, \dots, x_n, z\} + z)$.

ИС:

$$h(\bar{x}, y + 1) = f(\bar{x}, y, h(\bar{x}, y)) < F(k, \max\{x_1, \dots, x_n, y, h(\bar{x}, y)\})$$

По предположение $h(\bar{x}, y) < F(k+1, \max\{x_1, \dots, x_n, y\} + y)$, а също и $x_1, \dots, x_n, y < F(k+1, \max\{x_1, \dots, x_n, y\})$, тогава за горния израз получаваме:

$$F(k, \max\{x_1, \dots, x_n, y, h(\bar{x}, y)\}) < F(k, F(k+1, \max\{x_1, \dots, x_n, y\} + y)) =$$

$$F(k+1, \max\{x_1, \dots, x_n, y\} + y + 1) \leq F(k+1, \max\{x_1, \dots, x_n, y+1\} + (y+1))$$

Така получихме $h(x_1, \dots, x_n, y) < F(k+1, \max\{x_1, \dots, x_n, (y+1)\} + (y+1))$, с което индукционната стъпка е завършена.

Трябва обаче да докажем $P(h)$. Имаме, че $\max\{x_1, \dots, x_n, y\} + y \leq 2 \max\{x_1, \dots, x_n, y\}$. Достатъчно е да намерим такова $K \in \mathbb{N}$, че $\forall t : F(k+1, 2t) < F(K, t)$. Ползваме *Лема 4* и получаваме веригата:

$$h(\bar{x}, y) < F(k+1, \max\{x_1, \dots, x_n, y\} + y) \leq F(k+1, 2 \max\{x_1, \dots, x_n, y\}) \leq F(K, \max\{x_1, \dots, x_n, y\}) \checkmark$$

• **суперпозиция:** Нека $g_1, \dots, g_m \in \mathcal{F}_n, f \in \mathcal{F}_m$ са примитивно рекурсивни, като $P(g_1), \dots, P(g_m)$ и $P(f)$, а $h \in \mathcal{F}_n$ е получена чрез суперпозиция от тях. По предположение са в сила следните:

$$P(f) \Leftrightarrow \exists k_0 \forall \bar{y} : f(\bar{y}) < F(k_0, \max\{y_1, \dots, y_m\})$$

$$P(g_i) \Leftrightarrow \exists k_i \forall \bar{x} : g_i(\bar{x}) < F(k_i, \max\{x_1, \dots, x_n\}) \text{ за всяко } i, 1 \leq i \leq m$$

Отново ползваме, че функцията F е растяща по двата си аргумента, така че горните остават в сила и за $k := \max\{k_0, \dots, k_m\}$. Тогава

$$h(\bar{x}) = f(g_1(\bar{x}), \dots, g_m(\bar{x})) < F(k, \max\{g_1(\bar{x}), \dots, g_m(\bar{x})\})$$

Но $\max\{g_i(\bar{x}) \mid 1 \leq i \leq m\} < \max\{F(k, \max\{x_1, \dots, x_n\}) \mid 1 \leq i \leq m\} = F(k, \max\{x_1, \dots, x_n\}) \leq F(k+1, \max\{x_1, \dots, x_n\})$. Сега ползваме монотонността на F и *Лема 3*:

$$h(\bar{x}) < F(k, \max\{g_1(\bar{x}), \dots, g_m(\bar{x})\}) \leq F(k, F(k+1, \max\{x_1, \dots, x_n\})) =$$

$$F(k+1, \max\{x_1, \dots, x_n\} + 1) \leq F(k+2, \max\{x_1, \dots, x_n\})$$

Горното доказва, че суперпозицията на примитивно рекурсивни функции запазва свойството $P(h)$. \checkmark

Ако допуснем, че функцията на Акерман е примитивно рекурсивна, то в сила е и $P(F)$, т.е. съществува k , за което $F(x, y) < F(k, \max\{x, y\})$ при всеки избор на x, y . При $x = y = k$ обаче имаме $F(k, k) < F(k, \max\{k, k\})$, противоречие. ■

Задача 3 (Канонично кодиране). Да означим с Fin множеството от всички крайни подмножества на \mathbb{N} . Дефинираме изображение $\kappa : Fin \rightarrow \mathbb{N}$ по следния начин: ако $A = \{x_1, \dots, x_n\}$, то $\kappa(A)$ е числото, в чиито двоичен запис единиците са точно на позиции x_1, \dots, x_n (броим позициите от дясно наляво, започвайки от 0). Да се докаже, че κ е примитивно рекурсивно кодиране.

Решение.

• **κ е биекция:** За начало да дефинираме κ в явен вид: $\kappa(A) = \sum_{x_i \in A} 2^{x_i}$. Ще покажем, че изображението е биективно, тоест $(\forall z \in \mathbb{N})(\exists! A \in Fin)(\kappa(A) = z)$.

Възможност това следва директно от факта, че всяко число има единствено представяне в двоична бройна система, но можем да докажем и с индукция по z .

База: За $z = 0 : \kappa(A) = 0 \Leftrightarrow \sum_{x_i \in A} 2^{x_i} \Leftrightarrow A = \emptyset$. \checkmark

ИП: Нека за всяко $t < z$ съществува единствено крайно множество, чийто код е t .

ИС: Нека k е най-малката позиция, на която има единица в z . Това означава, че ако съществува A , за което $\kappa(A) = z$, то със сигурност $k \in A$. Тогава $t = z - 2^k < z$ има същото двоично представяне като z с разликата, че на k -та позиция вече стои 0. От инд. предположение съществува единствено множество B , за което $\kappa(B) = t$. Тогава $A = B \cup \{k\}$ е единственото, за което е в сила $\kappa(A) = z$. \checkmark

Нека $z = \kappa(A)$. Дефинираме $lh(z) := |A|$ и

$$mem(z, i) = \begin{cases} i\text{-тия по големина елемент на } A, & \text{ако } 1 \leq i \leq lh(z) \\ 0, & \text{иначе.} \end{cases}$$

Ще покажем, че $lh(z), mem(z, i)$ са примитивно рекурсивни.

- **$lh(A)$ е примитивно рекурсивна:** Имаме, че $lh(z) = |A| = брой единици в двоичното представяне на z .$ Можем да ползваме това, че двоичното представяне на $\left[\frac{z}{2}\right]$ има цифрите на $z_{(2)}$, но преместени надясно с 1 позиция. На нулева позиция в $z_{(2)}$ пък стои 1 точно когато числото е нечетно. Това ни дава следната зависимост:

$$lh(z) = \begin{cases} lh(qt(z, 2)), & \text{ако } z > 0 \& rem(z, 2) = 0 \\ lh(qt(z, 2)) + 1, & \text{ако } z > 0 \& rem(z, 2) = 1 \\ 0, & \text{ако } z = 0 \end{cases}$$

Това можем да запишем и като $lh(z) = G(z, lh(qt(z, 2)))$ за $z > 0$. И понеже тогава $qt(z, 2) < z$, то примитивната рекурсивност на lh следва от схемата за силна рекурсия. \checkmark

Впрочем същото може да се види и директно от явния вид на функцията: $lh(z) = \sum_{i < z} rem(qt(z, 2^i), 2)$.

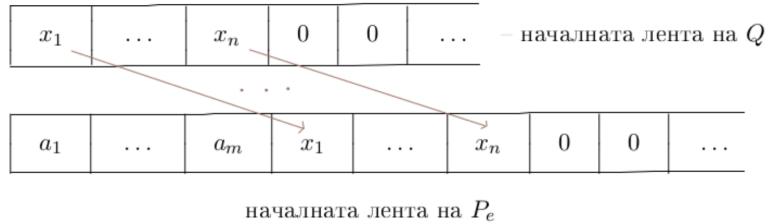
- **$mem(z, i)$ е примитивно рекурсивна:** Да отбележим, че x_i -тият бит от двоичното представяне на z е 1 точно когато $\left[\frac{z}{2^{x_i}}\right]$ е нечетно число, т.e. е $rem(qt(z, 2^{x_i}), 2)$. Тогава броят единици в $z_{(2)}$ до позиция j включително е $\sum_{k=0}^j rem(qt(z, 2^k), 2)$, което е примитивно рекурсивна функция (ограничена сума). Също така z има не повече от z на брой единици в двоичния запис. Така $mem(z, i) = \mu j_{j < z} [\underbrace{\sum_{k=0}^j rem(qt(z, 2^k), 2)}_{p(z, i, j)} = i]$ е примитивно рекурсивна, защото предикатът $p(x, y, z)$ също е. Исканото за κ следва. ■

Задача 4. Докажете, че S_n^m функцията в S_n^m -теоремата може да се избере така, че да не зависи от n . Тоест съществува функция S^m такава, че за всяко $n \geq 1$ и всички естествени $e, a_1, \dots, a_m, x_1, \dots, x_n$:

$$\varphi_{S^m(e, a_1, \dots, a_m)}^{(n)}(x_1, \dots, x_n) \simeq \varphi_e^{(n+m)}(a_1, \dots, a_m, x_1, \dots, x_n)$$

Решение. За начало е добре да отбележим как нашето доказателство на S_n^m -теоремата се влияе от n . Зависимостта се дължи на прехвърлянето на съдържанието на регистрите, по-конкретно в изместването им с m позиции надясно:

$$X_{m+n} := X_n, \dots, X_m + 1 := X_1$$



Хубавото е, че преместването винаги е с m позиции надясно. Проблемът обаче е, че ако искаме да обобщим доказателството за произволно n , не знаем колко регистъра да преместим. Всъщност знаем, ключовото наблюдение е, че не ни трябва целият вход x_1, \dots, x_n , ако програмата работи само с първите k регистъра от него. Остава да намерим алгоритично това k . Да означим с $maxreg$ функцията, която по подаден код на инструкция, връща номера на максималния регистър, който тя достъпва. Можем да изразим функцията:

$$maxreg(z) = \begin{cases} \frac{z}{4} + 1 & \text{ако } z = 4l \\ \left[\frac{z}{4}\right] + 1 & \text{ако } z = 4l + 1 \\ max\{L(\left[\frac{z}{4}\right]), R(\left[\frac{z}{4}\right])\} + 1 & \text{ако } z = 4l + 2 \\ max\{J_1^3(\left[\frac{z}{4}\right]), J_2^3(\left[\frac{z}{4}\right])\} + 1 & \text{ако } z = 4l + 3 \end{cases}$$

Ясно е, че maxreg е примитивно рекурсивна. Нека $P_e : I_0, I_1, \dots, I_t$. Да дефинираме и функция Maxreg , която по код на програма връща максималния достъпен регистър от нея, т.е. искаме $\text{Maxreg}(e) = \max\{\text{maxreg}(\beta(I_i)) \mid 1 \leq i \leq t\}$. Същата можем да запишем и така:

$$\text{Maxreg}(e) = \max_{0 \leq i \leq \text{lh}(e)} \text{maxreg}(\text{mem}(e, i))$$

Следователно Maxreg също е примитивно рекурсивна (понеже се получава с ограничена максимизация). Получихме $k := \text{Maxreg}(e)$. Промяната в доказателството е, че този път инструкциите на $Q_{e,\bar{a}}$ са:

$$\begin{aligned} X_{m+k} &:= X_m, \dots, X_{m+1} := X_1, \\ X_1 &:= 0, \underbrace{S(1), \dots, S(1)}_{a_1 \text{ пъти}}, \dots, X_m := 0, \underbrace{S(m), \dots, S(m)}_{a_m \text{ пъти}} \\ &\quad \underbrace{I'_0, I'_1, \dots, I'_t}_{\text{преадресираните инструкции на } P_e} \end{aligned}$$

Това се отразява на функцията $f(e, a, l)$ по следния начин:

$$f(e, a, l) = \begin{cases} \beta(T(k + l + 1, l + 1)), & \text{ако } l < k \\ \beta(Z(1)), & \text{ако } l = k \\ \beta(S(1)), & \text{ако } k + 1 \leq l < a_1 + k + 1 \\ \dots \\ \text{prim}(\text{mem}(e, l - k - m - \sum_{i \leq m} a_i)), & \text{ако } a_1 + \dots + a_m + m + k \leq l \leq a_1 + \dots + a_m + m + k + \text{lh}(e) \\ 0, & \text{ако } l > a_1 + \dots + a_m + k + \text{lh}(e). \end{cases}$$

Функцията на f продължава да бъде примитивно рекурсивна. Останалата част от доказателството остава същата. ■

Задача 5. Докажете, че при фиксиранни $n, m \geq 1$ съществуват безброй много S_n^m функции, които могат да се генерират ефективно, т.е. съществува рекурсивна[†] функция h такава, че $h(k)$ е индекс на някоя S_n^m функция за всяко k .

Решение. Нека функция f се пресмята от програмата $P : I_0, \dots, I_k$. Ясно е, че същата функция би се пресмятала и от програмата $Q : I_0, \dots, I_k, X_1 := X_1$. Това обаче задава друг индекс на същата функция. Можем да се възползваме от факта и да конструираме друга S_n^m функция. Имаме, че

$$\gamma(P) = \tau(\langle \beta(I_0), \dots, \beta(I_k) \rangle)$$

$$\gamma(Q) = \tau(\langle \beta(I_0), \dots, \beta(I_k), \beta(X_1 := X_1) \rangle)$$

Имаме предвид, че $\tau(\langle x_0, \dots, x_n \rangle) = \Pi(n, \Pi_{n+1}(x_0, \dots, x_n))$. Тогава кода на втората програма можем да изразим чрез този на първата по следния начин:

$$\gamma(Q) = \underbrace{\Pi(\text{lh}(\gamma(P)) + 1, \Pi(R(\gamma(P)), \beta(X_1 := X_1)))}_{G(\gamma(P))}$$

Функцията $G(x)$ е примитивно рекурсивна. Нейната цел е по подаден индекс на функция да върне друг индекс на същата функция.

S_m^n -теоремата гарантира съществуване на поне една S_m^n функция. Нека въпросната има индекс е s_0 . Дефинираме функция $F(n, e, a_1, \dots, a_m)$ по следния начин:

$$\left| \begin{array}{l} F(0, e, \bar{a}) = S_n^m(e, \bar{a}) \\ F(k + 1, e, \bar{a}) = G(F(k, e, \bar{a})) \end{array} \right.$$

Функцията F е получена с примитивна рекурсия от примитивно рекурсивни, следователно и тя е такава. Да забележим, че дефиницията ни на G по-горе, гарантира $\varphi_{F(n,e,\bar{a})}^{(n)} = \varphi_{F(n+1,e,\bar{a})}^{(n)}$ (докато

[†]в доказателството показваме нещо по-силно, че има примитивно рекурсивна, което ме навежда на мисълта, че може да има грешка :\

$$F(n, e, \bar{a}) \neq F(n + 1, e, \bar{a}).$$

Понеже $F(n, e, a)$ е изчислима, за нея можем да ползваме S_n^m -теоремата (в "слабия" вариант), което ни гарантира съществуване на примитивно рекурсивна функция h , за която:

$$\varphi_{h(k)}^{(m+1)}(e, \bar{a}) = F(k, e, \bar{a})$$

Твърдим, че $h(k)$ е индекс на S_m^n функция за всяко k . Достатъчно е да проверим

$$\varphi_{\varphi_{h(k)}^{(m+1)}(e, a)}^{(n)}(\bar{x}) = \varphi_{F(k, e, a)}^{(n)}(\bar{x}) = \varphi_{F(0, e, a)}^{(n)}(\bar{x}) = \varphi_{S_n^m(e, a)}^{(n)}(\bar{x})$$

Това доказва исканото. ■

Задача 6. Докажете, че съществува примитивно рекурсивна функция $c(n, a)$ такава, че за всеки две естествени числа n, a е в сила $\varphi_{c(n, a)} = C_a^n$.

Решение. [†]За фиксирано a разглеждаме програмата: $P_e : Z(1), \underbrace{S(1), \dots, S(1)}_{a \text{ пъти}}$. Ясно е, че за всеки n и $\bar{x} : \varphi_e^{(n)}(\bar{x}) = a$. Дефинираме функцията $c(n, a)$ по следния начин:

$$\begin{cases} c(n, 0) = \tau(\langle \beta(Z(0)) \rangle) \\ c(n, a + 1) = \Pi(a + 2, \Pi(R(c(n, a)), \beta(Z(1)))) \end{cases}$$

Въпросната функция е очевидно примитивно рекурсивна, при това за всеки n, a :

$$c(n, a) = \tau(\langle \beta(Z(1)), \underbrace{\beta(S(1)), \dots, \beta(S(1))}_{a \text{ пъти}} \rangle)$$

Тогава автоматично $\varphi_{c(n, a)}^{(n)}(x_1, \dots, x_n) = a = C_a^n(\bar{x})$. ■

Задача 7. Докажете, че съществува рекурсивна функция g такава, че за всяко n числото $g(n)$ е индекс на функцията g^n .

Решение. Ще ползваме следното помошно твърдение (което впрочем е доказвано):

Лема 5

Съществува двуместна примитивно рекурсивна функция p , такава че за всички естествени n и a .

$$\varphi_{p(a, n)} = \varphi_a^n$$

Доказателство. Нека $F(n, a, x) \simeq \varphi_a^n(x)$. Същата може да се изрази от следната примитивно рекурсивна схема:

$$\begin{cases} F(0, a, x) = x \\ F(n + 1, a, x) \simeq \Phi(a, F(n, a, x)) \end{cases}$$

Следва, че F е изчислима, тогава и $P(a, n, x) = F(n, a, x)$ също е. По S_n^m -теоремата съществува примитивно рекурсивна функция $p(a, n)$, за която:

$$\varphi_{p(a, n)}(x) \simeq G(a, n, x) \simeq \varphi_a^n(x)$$

□

Нека такава функция g съществува и a е неин индекс. Тогава трябва да е в сила следното:

$$\varphi_{\varphi_a(n)}(x) \simeq \underbrace{\varphi_a^n(x)}_{f(a, n, x)}$$

[†]За съжаление това решение излезе доста безидейно :(

Можем да изразим

$$f(a, n, x) \simeq \Phi_1(p(a, n), x)$$

Следователно f е изчислима. Прилагаме S_n^m -теоремата за нея и получаваме, че съществува примитивно рекурсивна функция h , за която при всеки a, n, x :

$$\varphi_{h(a,n)}(x) \simeq f(a, n, x)$$

Тогава е достатъчно е да докажем, че съществува индекс a , за който

$$\varphi_{\varphi_a(n)}(x) \simeq \varphi_{h(a,n)}(x)$$

Можем директно да приравним $\varphi_a(n) = h(a, n)$. *Теоремата за определимост по рекурсия* гарантира съществуването на такова a . В частност, $g = \varphi_a(n)$ е рекурсивна. ■

Задача 8. Докажете, че съществува инективна и рекурсивна функция g такава, че за всяко n числото $g(n)$ е индекс на g .

Решение. По-нагоре показвахме, че съществува примитивно рекурсивна функция $nxt(e)$, която по код на програма връща код на еквивалентна програма със стриктно повече инструкции от първата (идеята беше да добавя празна инструкция $T(1, 1)$ в края). Това гарантира, че за всеки $a, b \in \mathbb{N}$: $nxt^a(e) = nxt^b(e) \implies a = b$, т.e. при фиксирано e функцията $g(n) := nxt^n(e)$ е инективна. Тогава $\forall a, b : \varphi_{g(a)} = \varphi_{g(b)}$. Достатъчно е да докажем, че съществува такова e , за което $g(0) = nxt^0(e) = e$ е индекс на g . Последното можем да изразим и така:

$$\varphi_e(n) = \underbrace{nxt^n(e)}_{f(e,n)}$$

Ясно е, че $f(e, n)$ е изчислима (това по същество е итерацията на функцията nxt). Тогава по *Теоремата за определимост по рекурсия* съществува e , за което $\varphi_e(n) = f(e, n) = nxt^n(e) = g(n)$. Тогава

$$\varphi_{g(n)}(x) = \varphi_{g(0)}(x) = \varphi_e(x) = g(n) \quad \blacksquare$$

Задача 9. Измислете никакво автореферентно свойство на програма за МНР и докажете, че има безброй много програми с това свойство. Да се докаже, че съществуват безброй много програми P_e , за които е в сила:

$$P_e \text{ спира върху вход } x \iff P_x \text{ спира върху вход } e$$

Решение. Достатъчно е да докажем, че съществуват безброй много e такива, че за всяко x

$$\varphi_e(x) \simeq \varphi_x(e)$$

Дефинираме функция $f(e, x) \simeq \varphi_x(e) \simeq \Phi_1(x, e)$. Следователно f е изчислима. Тогава по *Теоремата за определимост по рекурсия* следва, че съществува e такова, че

$$\varphi_e(x) \simeq f(e, x) \simeq \varphi_x(e)$$

Доказвали сме, че щом съществува едно, то съществуват безброй много e с въпросното свойство (следва от факта, че съществува безброй много псевдонеподвижни точки). ■

Задача 10. Нека $\emptyset \neq A \subseteq \mathbb{N}$ е разрешимо. Конструирайте чрез χ_A рекурсивна нестрого растяща функция h такава, че $A = \{h(0), h(1), \dots\}$.

Решение. Дефинираме h по следния начин:

$$\left| \begin{array}{l} h(0) = \mu x[\chi_A(x) = 0] \\ h(x+1) = H(x, h(x)), \end{array} \right.$$

където $H(x, y) = \begin{cases} x+1, & \text{ако } \chi_A(x+1) = 0 \\ y, & \text{иначе} \end{cases}$

• **рекурсивност:** От горните е видно, че така построената h е частично рекурсивна. Функцията H е рекурсивна, защото е построена с разглеждане на случаи от примитивно рекурсивни функции и рекурсивни предикати. При това непразнотата на A и тоталността на χ_A гарантират, че $!\mu x[\chi_A(x) = 0]$, следователно h е рекурсивна. ✓

• **монотонност:** Сега да се убедим, че h е нестрого растяща. Нека $m := \min\{x|x \in A\}$. Тогава е ясно, че $h(0) = m$ (понеже $\chi_A(m) = 0$, а $\chi_A(x) = 1$ за $x < m$). Макар че е видно, че докажем по индукция, че за $x \leq m : h(x) = m$. Базовият случай е $x = 0$, вече показахме. За $0 \leq x < m$ по дефиниция имаме

$$h(x+1) = H(x, h(x)) = \begin{cases} x+1, & \text{ако } \chi_A(x+1) = 0 \\ h(x), & \text{иначе} \end{cases}$$

От индуктивната хипотеза знаем, че $h(x) = m$. От избора на m пък следва, че:

- За $x+1 < m : \chi_A(x+1) = 1$, така че попадаме във втория случай от дефиницията на H и така $h(x+1) = H(x, h(x)) = h(x) = m$.
- За $x+1 = m$ пък знаем $\chi_A(m) = 0$, така че сме в първия случай и отново $h(x+1) = m$. В крайна сметка получихме $h(0) = h(1) = \dots = h(m) = m$.

Сега да покажем монотонността за следващите елементи в редицата. Доказваме по индукция, че за всяко $x \geq m$ е в сила $P(x) \Leftrightarrow h(x) \leq x$. Базовият случай вече бе показан. За $x \geq m$ имаме:

$$h(x+1) = H(x, h(x)) = \begin{cases} x+1 (\leq x+1), & \text{ако } \chi_A(x+1) = 0 \\ h(x) (\leq x \text{ по хипотеза}), & \text{иначе} \end{cases}$$

И в двата случая $h(x+1) \leq x+1$. Вече за всяко $x \geq m$ се вижда, че

$$h(x+1) = H(x, h(x)) = \begin{cases} x+1 (> x \geq h(x)), & \text{ако } \chi_A(x+1) = 0 \\ h(x), & \text{иначе} \end{cases}$$

И в двата случая $h(x+1) \geq h(x)$, с което монотонността е доказана. ✓

• **коректност:**

($A \subseteq \text{Range}(h)$): Вече споменахме, че $h(0) = \dots = h(m) = m \in A$. За $x+1 > m$ доказателството отново протича по индукция. Ако $\chi_A(x+1) = 0$, то $x+1 \in A$, а тогава влизаме и в първия случай от дефиницията на H , значи $h(x+1) = x+1 \in A$. ✓

($A \subseteq \text{Range}(h)$): Ясно е, че $m = h(0) \in A$. Нека сега $a \in A$ е друг елемент от множеството ($a \neq m$). Минималността на m гарантира $a > m \geq 0$, значи $a \geq 1$. Също така $\chi_A(a) = 0 \Rightarrow h(a) = h((a-1)+1) = a$. Следователно $a \in \text{Range}(h)$. ✓

Получихме, че $A = \text{Range}(h)$ и h е рекурсивна, нестрого растяща функция. ■

Задача 11. Нека f и g са едноместни рекурсивни функции, като g е биективна. Нека още $f(x) \geq g(x)$ за всяко $x \in \mathbb{N}$. Докажете, че $\text{Range}(f)$ е разрешимо множество.

Решение. Ще започнем с доказването на следното помошно твърдение.

Лема 6

Ако $g : \mathbb{N} \rightarrow \mathbb{N}$ е рекурсивна (тотална и изчислима) и биективна функция, то обратната ѝ функция $g^{-1}(x)$ също е рекурсивна.

Доказателство. Дефинираме $g^{-1}(y) = \mu x[g(x) = y]$, което можем да запишем и като

$$g^{-1}(y) = \mu x \underbrace{[g(x) - y]}_{G(x,y)} = 0$$

Функцията $g^{-1}(x)$ се получава с минимизация от рекурсивната $G(x,y)$, така е частично рекурсивна. Фактът, че g е биекция гарантира съществуването на x с необходимото свойство за всяко y , а оттам и тоталността на g^{-1} . \square

Обратно към задачата. За начало можем да използваме, че $f(x) \geq g(x)$, така че, ако $g(x) > y$ е ясно, че $f(x) \neq y$. Хубавото е, че само за краен брой стойности на $x : g(x) \leq y$ (това се гарантира от биективността на g). Всъщност достатъчно е да ограничим “търсенето” до $b_y = \max\{x \in A | g(x) \leq y\}$. Оттам нататък, т.е. за $x > b_x : f(x) \geq g(x) > y$. Дефинираме функция $b(y)$ по следния начин:

$$\begin{cases} b(0) = g^{-1}(0) \\ b(y+1) = \max(b(y), g^{-1}(y+1)) \end{cases}$$

Ясно е, че b е рекурсивна, защото се получава чрез примитивна рекурсия от рекурсивни функции. По индукция можем да докажем, че $b(y) = \max\{x \in \mathbb{N} | g(x) \leq y\}$. Базата е $y = 0$, имаме $b(0) = g^{-1}(0)$, което е единственото (в частност максималното) x с $g(x) \leq 0$. Понеже за всяко y имаме единствено x , за което $g(x) = y$, то е в сила $\max\{x \in \mathbb{N} | g(x) \leq y+1\} = \max\{\max\{x \in \mathbb{N} | g(x) \leq y\}, g^{-1}(y+1)\}$, което и имаме в дефиницията на b .

Можем да разгледаме два случая:

- **Range(f) е крайно:** Директно следва, че то и разрешимо, характеристичната му функция получаваме с изброяване на случаи:

$$\chi(z) = \begin{cases} 0, & \text{ако } z = y_1 \\ \dots \\ 0, & \text{ако } z = y_k \\ 1, & \text{иначе} \end{cases}$$

- **Range(f) е безкрайно:** Тогава то е и неограничено. Използваме, че $\text{Range}(f)$ е разрешимо тогава и само тогава, когато съществува строго растяща рекурсивна функция $h : \mathbb{N} \rightarrow \mathbb{N}$ избройтел на множеството, т.е. $\text{Range}(f) = \text{Range}(h) = \{h(0), h(1), \dots\}$. Ще конструираме такава, която ще изглежда горе-долу така[†]:

$$\begin{cases} h(0) = \mu y[\exists x f(x) = y] \\ h(n+1) = \underbrace{\mu y[\exists x f(x) = y \& y > h(n)]}_{F(n,h(n))} \end{cases}$$

Това е по-скоро отправна точка. Проблемът обаче е в неограничеността на екзистенциалните квантори. Ето защо ще ги ограничим. Ползваме направените по-горе наблюдения за $b(x)$:

$$\begin{cases} h(0) = \mu y[\exists x_{x \leq b(y)} f(x) = y] \\ h(n+1) = \underbrace{\mu y[\exists x_{x \leq b(y)} f(x) = y \& y > h(n)]}_{F(n,h(n))} \end{cases}$$

[†]Забележка: Впрочем дефинирането на такава функция се оказа излишно. Доста по-лесно е директно да ползваме НДУ за разрешимост на множеството, след като вече имаме $b(x)$:

$$y \in \text{Range}(f) \Leftrightarrow \exists x_{x \leq b(y)} f(x) = y \Leftrightarrow \prod_{x \leq b(y)} |f(x) - y| = 0$$

Изчислимостта на h следва от това, че предикатът е рекурсивен. Функцията f е неограничена, което гарантира $\exists y[\exists x_{x \leq b(y)} f(x) = y \& y > h(n)]$ тоталността на h . Заключаваме, че h е рекурсивна. Монотонността на h също се вижда от дефиницията $/h(n+1) = \mu y[\dots \& y > h(n)]/$.

Пак от там следва, че $\text{Range}(h) \subseteq \text{Range}(f)$. Да допуснем обаче, че $\text{Range}(f) \not\subseteq \text{Range}(h)$ и съществува число y , за което $y \in \text{Range}(f) \& y \notin \text{Range}(h)$. Нека y_0 е най-малкото с това свойство. Нека $n_0 = \max\{n \in \mathbb{N} | h(n) < y_0\}$. Тогава $h(n_0 + 1) > y_0$, но същевременно $\exists x f(x) = y_0$ и $y_0 > h(n_0)$, тоест y_0 изпълнява условията на минимизацията и така $h(n + 1) = y_0$ което е противоречие. Това доказва и обратното включване. Следва, че $\text{Range}(f) = \text{Range}(h)$, което пък окончателно доказва разрешимостта на $\text{Range}(f)$. ■

Задача 12. Нека f и g са едноместни рекурсивни функции, като g е обратима (инективна) и $\text{Range}(g)$ е разрешимо. Нека още $f(x) \geq x$ за всяко $x \in \mathbb{N}$. Докажете, че е разрешимо множеството $S := \{g(x) | x \in \text{Range}(f)\}$.

Решение. Ще използваме наготово (доказано на лекции), че при тези условия множеството $\text{Range}(f)$ е разрешимо. Нека неговата характеристична функция е χ_f , а тази на $\text{Range}(g)$ съответно χ_g . Дефинираме функцията

$$g^{-1}(y) \simeq \begin{cases} \mu x[|g(x) - y| = 0] + 1, & \text{ако } y \in \text{Range}(g) \\ 0, & \text{иначе} \end{cases}$$

Предикатът $y \in \text{Range}(g)$ е рекурсивен (с характеристична функция χ_g), а минимизацията и константата са частични функции, следователно $g^{-1}(y)$ е частично рекурсивна. Тук трябва да направим някои уточнения - ползваме `if then else` конструкцията, която поддържа лениво оценяване (call by name). Това гарантира, че при $y \in \text{Range}(g)$ е вярно $\exists x [f(x) = y]$, а оттук и че g^{-1} е тотална. Заключаваме, че g^{-1} е рекурсивна.

Можем да запишем условието за принадлежност към S по следния начин:

$$y \in S \Leftrightarrow y \in \text{Range}(g) \& (g^{-1}(y) \dot{-} 1) \in \text{Range}(f) \Leftrightarrow \chi_g(y) + \chi_f(g^{-1}(y) \dot{-} 1) = 0$$

Функцията вдясно е рекурсивна, следователно по НДУ за разрешимост на множество това множество е разрешимо. ■

Задача 13. Селектор за $A \subseteq \mathbb{N}^{n+1}$ наричаме всяка n -местна функция n , за която е изпълнено:

$$\exists y (\bar{x}, y) \in A \rightarrow !f(x) \& (\bar{x}, f(\bar{x})) \in A$$

Докажете, че всяко полуразрешимо множество $A \subseteq \mathbb{N}^n, n \geq 2$ притежава изчислим селектор.

Решение. Всъщност това, което трябва да докажем, доста напомня едната посока на *Теорема за графиката*. Доказателството е аналогично.

От НДУ за полуразрешимост следва, че съществува рекурсивна функция ρ :

$$A = \{(\bar{x}, y) | \exists t \rho(\bar{x}, y, t) = 0\}$$

Дефинираме $f(x) \simeq L(\mu z[\rho(\bar{x}, L(z), R(z)) = 0])$. Изчислимостта на f следва от дефиницията. Твърдим, че така дефинираната е и селектор.

коректност: Нека $(\bar{x}, y) \in A$. Тогава съществува $t : \rho(\bar{x}, y, t) = 0$. Да означим $z := \Pi(y, t)$. Тогава $\rho(\bar{x}, L(z), R(z)) = 0$. Това, комбинирано с факта, че ρ е рекурсивна, гарантира, че $\mu z[\rho(\bar{x}, L(z), R(z)) = 0]$, откъдето и $\underline{!f(\bar{x})}$. Тогава

$$(\bar{x}, f(\bar{x})) = (\bar{x}, L(\mu z[\rho(\bar{x}, L(z), R(z)) = 0])),$$

като нека означим $z_0 := \mu z[\rho(\bar{x}, L(z), R(z)) = 0]$. Знаем, че $f(\bar{x}) = L(z_0)$. И понеже $\rho(x, L(z_0), R(z_0)) = 0$, то $\rho(\bar{x}, f(\bar{x}), R(z_0)) = 0$, откъдето $\underline{(\bar{x}, f(\bar{x})) \in A}$. ■

Задача 14. Нека A е разрешимо. Докажете, че следните две условия са еквивалентни:

- 1) A има рекурсивен селектор;
- 2) $B = \{ \bar{x} \mid \exists y (\bar{x}, y) \in A \}$ е разрешимо.

Решение.

Нека χ_A е характеристичната функция на A (тя е рекурсивна).

1) \Rightarrow 2): Нека f е рекурсивен селектор за множеството A . Лесно се вижда, че

$$\bar{x} \in B \Leftrightarrow (\bar{x}, f(\bar{x})) \in A$$

Обратната посока следва от дефиницията на B . В правата посока твърдението също е в сила, защото

$$\bar{x} \in B \Leftrightarrow \exists y (\bar{x}, y) \in A \Rightarrow (\bar{x}, f(\bar{x})) \in A$$

В крайна сметка

$$\bar{x} \in B \Leftrightarrow (\bar{x}, f(\bar{x})) \in A \Leftrightarrow \chi_A(\bar{x}, f(\bar{x})) = 0$$

Последната функция е рекурсивна, следователно по *HДУ за разрешимост* на множество B е разрешимо. ✓

2) \Rightarrow 1): Дефинираме f по следния начин

$$f(\bar{x}) = \begin{cases} 1, & \text{ако } \bar{x} \notin B \\ \mu y [\chi_A(\bar{x}, y) = 0], & \text{иначе} \end{cases}$$

Първо да се уверим, че функцията е рекурсивна. Това следва от факта, че χ_A е такава (следователно $!\chi_A(\bar{x}, y)$), и от $\bar{x} \in B \Leftrightarrow \exists y (\bar{x}, y) \in A \Leftrightarrow \exists y \chi_A(\bar{x}, y) = 0$.

Нека сега $\exists y (\bar{x}, y) \in A$, тогава $\bar{x} \in B$ и както стана ясно от горното, $!f(x)$. В такъв случай $f(x) = y_0 = \mu y [\chi_A(\bar{x}, y) = 0]$. Следователно $\chi_A(\bar{x}, y_0) = 0$ и наистина

$$(\bar{x}, f(\bar{x})) = (\bar{x}, y_0) \in A$$

Това доказва, че рекурсивната f е селектор. ■