# МЛТА. Лекція 07.06.2021

Всі пари натуральних чисел розташуємо в послідовність так, що пара $(x,y)$ йде раніше пари $(u,v)$, якщо $x+y<u+v$ або $x+y=u+v$ та $x<u$.

Отримаємо таку послідовність пар натуральних чисел:

$$(0,0),\ (0,1),\ (1,0),\ (0,2),\ (1,1),\ (2,0),\ (0,3),\ldots$$

Номер пари $(x,y)$ в такій послідовності позначають $C(x,y)$ і називають ***канторовським номером*** пари $(x,y)$, причому нумерацію починають з нуля.

Отже,

$$C(0,0)=0,\ C(0,1)=1,\ C(1,0)=2,\ C(0,2)=3,\ C(1,1)=5,\ C(2,0)=5,\ \ldots$$

Ліву та праву компоненти пари з номером $n$ позначимо відповідно $l(n)$ та $r(n)$:

$$l(n)=x,\ r(n)=y.$$

Функції $l(n)$ і $r(n)$ називаються ***лівою*** та ***правою координатними функціями***. Покажемо, що функції $C(x,y)$, $l(n)$ і $r(n)$ записуються через звичайні арифметичні функції.

**Теорема.** Функції $C(x,y)$, $l(n)$, $r(n)$ – примітивно рекурсивні.

Доведення.

Пара $(x,y)$ знаходиться на $x$-ому місці за парою $(0,x+y)$:

$$(0,x+y),\ (1,x+y-1),\ \ldots,\ (x,y),\ \ldots,\ (x+y,0),\ \ldots$$

Перед парою $(0,x+y)$ знаходиться $x+y$ груп пар з однаковою сумою компонент, причому в групі з сумою компонент $m$ міститься $m+1$ пара. Тому перед парою $(0,x+y)$ знаходиться усього

$$1+2+\ldots+(x+y)=\frac{(x+y+1)(x+y)}{2} \text{ пар.}$$

Тому

$$C(x,y)=x+\frac{(x+y+1)(x+y)}{2}=\frac{2x+(x+y+1)(x+y)}{2}=\frac{(x+y)^2+3x+y}{2},$$

$$n=C(x,y)=\left[\frac{(x+y)^2+3x+y}{2}\right]. \tag{1}$$

Отже, функція $C(x,y)$ – примітивно рекурсивна.

Знайдемо залежність $x$ і $y$ від $n$. З формули (1) маємо:

$$2n=(x+y)^2+3x+y.$$

Помножимо останню рівність на 4 і додамо 1:

$$8n+1=4(x+y)^2+12x+4y+1=(2x+2y+1)^2+8x=(2x+2y+3)^2-8y-8.$$

Звідси випливає

$$2x+2y+1 \le \left[\sqrt{8n+1}\right] < 2x+2y+3$$

або, додавши до нерівності 1 і розділивши на 2, отримаємо

$$x+y+1 \le \frac{\left[\sqrt{8n+1}\right]+1}{2} < x+y+2.$$

Отже,

$$x+y+1 = \left[\frac{\left[\sqrt{8n+1}\right]+1}{2}\right].$$

З формули (1) маємо

$$l(n) = x = n \dot{-} \left[\frac{(x+y+1)(x+y)}{2}\right] = n \dot{-} \frac{1}{2}\left[\frac{\left[\sqrt{8n+1}\right]+1}{2}\right] \cdot \left[\frac{\left[\sqrt{8n+1}\right]\dot{-}1}{2}\right]. \qquad (2)$$

Функція $r(n)$

$$r(n) = y = (x+y+1) \dot{-} (x+1) = \left[\frac{\left[\sqrt{8n+1}\right]+1}{2}\right] \dot{-} \left(l(n)+1\right). \qquad (3)$$

Отже, функції $l(n)$ та $r(n)$ – примітивно рекурсивні. Теорему доведено.

За допомогою нумерації пар натуральних чисел легко отримати нумерацію трійок, четвірок і інших множин натуральних чисел. Для цього вводимо такі функції:

$$C^3(x_1,x_2,x_3) = C\left(C(x_1,x_2),x_3\right),$$

$$C^4(x_1,x_2,x_3,x_4) = C^3\left(C(x_1,x_2),x_3,x_4\right) = C\left(C\left(C(x_1,x_2),x_3\right),x_4\right), \qquad (4)$$

… … … … … … … … … … … … …

$$C^{n+1}(x_1,x_2,\ldots,x_{n+1}) = C^n\left(C(x_1,x_2),x_3,\ldots,x_{n+1}\right).$$

Для $m = C^n(x_1,x_2,\ldots,x_n)$ можна визначити координатні функції $C_{n1}(m) = x_1$, $C_{n2}(m) = x_2$, …, $C_{nn}(m) = x_n$.

**Теорема.** Функції $C^n$, $C_{n1}$, …, $C_{nn}$ – ПРФ.

Без доведення.

## Геделівські нумерації

Розглянемо геделівські нумерації (*Kurt Friedrich Gödel*) послідовностей натуральних чисел, МНР-програм, програм МТ, ЧРФ.

### Геделівська нумерація послідовності натуральних чисел

Однозначну геделівську нумерацію всіх скінченних послідовностей натуральних чисел задаємо за допомогою такого кодування скінченних послідовностей:

$$\upsilon(a_1,a_2,\dots,a_n)=2^{a_1}+2^{a_1+a_2+1}+\dots+2^{a_1+a_2+\dots+a_n+n-1}-1.$$

Приклади:

$$\upsilon(3,1,0,2)=2^3+2^{3+1+1}+2^{3+1+0+2}+2^{3+1+0+2+3}-1=8+32+64+512-1=615,$$

$$\upsilon(4,3,2)=2^4+2^{4+3+1}+2^{4+3+2+2}-1=16+256+2048-1=2319,$$

$$\upsilon(5,3)=2^5+2^{5+3+1}-1=32+512-1=543,$$

$$\upsilon(3,5)=519,\ \upsilon(0)=0.$$

**Геделівська нумерація МНР-програм**

Однозначну геделівську нумерацію всіх МНР-програм задаємо на основі кодування МНР-програм як скінченних послідовностей команд МНР. Кодування команд МНР задаємо так:

$\kappa(Z(n))=4\cdot n$,

$\kappa(S(n))=4\cdot n+1$,

$\kappa(T(m,n))=4\cdot C(m,n)+2$,

$\kappa(J(m,n,q+1))=4\cdot C(C(m,n),q)+3$.

Якщо МНР-програма складається з послідовності команд $P=I_1I_2\dots I_k$, то геделів номер МНР-програми $P$ обчислюється так:

$$\tau(P)=\upsilon(\kappa(I_1),\kappa(I_2),\dots,\kappa(I_k)).$$

Для кожного числа $n\in N$ ефективно визначається МНР-програма $P=\varphi(n)$.

Спочатку представимо число $n+1$ як суму зростаючих ступенів двійки:

$$n+1=2^{b_1}+\dots+2^{b_k}\text{, де } 0\le b_1<\dots<b_k.$$

Далі визначаємо послідовність

$$a_1=b_1,\ a_{i+1}=b_{i+1}-b_i-1\text{, де } 1\le i<k.$$

За числами $a_1$, …, $a_k$ як за кодами команд МНР відновлюємо самі команди. Послідовність цих команд і є шуканою програмою МНР.

**Приклад 1**. Знайти код МНР-програми $P$, яка обчислює функцію $f(x,y)=x+y$.

$$n=C(x,y)=\left[\frac{(x+y)^2+3x+y}{2}\right]$$

1) $J(2,1,5)$ $\quad\kappa(J(2,1,5))=4\cdot C(C(2,1),4)+3=$
$=4\cdot C(8,4)+3=4\cdot 86+3=347$

2) $S(0)$ $\quad\kappa(S(0))=4\cdot 0+1=1$

3) $S(2)$ $\quad\kappa(S(2))=4\cdot 2+1=9$

4) $J(0,0,1)$ $\quad\kappa(J(0,0,1))=4\cdot C(C(0,0),0)+3=$

$$= 4 \cdot C(0,0) + 3 = 4 \cdot 0 + 3 = 3$$

Тоді

$$\tau(P) = \upsilon(347,1,9,3) = 2^{347} + 2^{347+1+1} + 2^{347+1+9+2} + 2^{347+1+9+3+3} - 1 =$$
$$= 2^{347} + 2^{349} + 2^{359} + 2^{363} - 1 = 1.996289864090458 \cdot 10^{109}.$$

**Приклад 2**. Відновити МНР-програму, якщо $P = \varphi(5119)$.

Представимо число $5119 + 1 = 5120$ як суму зростаючих ступенів двійки:

$$5120 = 2^{10} + 2^{12}, \text{ де } b_1 = 10,\ b_2 = 12.$$

Далі визначаємо послідовність

$$a_1 = b_1 = 10,\ a_2 = b_2 - b_1 - 1 = 12 - 10 - 1 = 1.$$

За числами $a_1$, $a_2$ як кодам команд МНР відновлюємо самі команди.

| | | |
|---|---|---|
| $10 = 4 \cdot 2 + 2 = 4 \cdot C(1,0) + 2$ | 1) $T(1,0)$ | Наприклад, |
| $1 = 4 \cdot 0 + 1$ | 2) $S(0)$ | $f(x,y) = y + 1$ |

**Геделівська нумерація МТ**

Геделівська нумерація усіх МТ задається на основі кодування МТ. Кожну МТ можна задати послідовністю команд так, щоб перша команда містила в лівій частині $q_1$, а остання команда містила в правій частині $q_0$. Множину команд МТ можна впорядкувати як послідовність зазначеного вигляду багатьма способами, тому наша нумерація МТ неоднозначна.

Кодування команд МТ задаємо так:

$$\mu(q_i a_j \to q_k a_l S) = 3 \cdot C^4(i,j,k,l),$$
$$\mu(q_i a_j \to q_k a_l L) = 3 \cdot C^4(i,j,k,l) + 1,$$
$$\mu(q_i a_j \to q_k a_l R) = 3 \cdot C^4(i,j,k,l) + 2.$$

Геделів номер МТ-програми $P = I_1 I_2 \ldots I_k$ обчислюється так:

$$\rho(P) = \upsilon(\mu(I_1), \mu(I_2), \ldots, \mu(I_k)).$$

**Приклад**. Обчислимо код МТ, яка обчислює функцію $f(x,y) = x + y + 2$

.

$I_1$: 1) $q_1 1 q_1 1 R$

$I_2$: 2) $q_1 0 q_0 1 S$

Нехай $a_0 = 0$, $a_1 = 1$ – символи алфавіту. Нагадаємо також формулу для обчислення $C(x,y)$:

$$C(x,y) = \left[ \frac{(x+y)^2 + 3x + y}{2} \right].$$

$$\mu(I_1) = \mu(q_1 1 q_1 1 R) = 3 \cdot C^4(1,1,1,1) + 2 = 3 \cdot C^3(C(1,1),1,1) + 2 =$$
$$= 3 \cdot C(C(4,1),1) + 2 = 3 \cdot C(19,1) + 2 = 3 \cdot 229 + 2 = 689.$$
$$\mu(I_2) = \mu(q_1 0 q_0 1 S) = 3 \cdot C^4(1,0,0,1) = 3 \cdot C^3(C(1,0),0,1) =$$

$= 3 \cdot C\left(C\left(2{,}0\right),1\right) = 3 \cdot C\left(5{,}1\right) = 3 \cdot 26 = 78$.

Тоді

$$\tau\left(P\right) = \upsilon\left(689{,}78\right) = 2^{689} + 2^{689+78+1} - 1 = 2^{689} + 2^{768} - 1 = 1.552518092300709 \times 10^{231}$$

## Геделівська нумерація ЧРФ

Геделівська нумерація усіх ЧРФ задається на основі кодування операторних термів алгебри ЧРФ.

Нехай алфавіт складається з символів базисних функцій $O$, $S$, $I_n^m$, символів операцій $R$, $M$, $S^{n+1}$, де $n > 0$, і допоміжних символів «(», «)», «.».Тоді має місце таке індуктивне визначення операторного терма (ОТ):

1. Кожен символ базисної функції – ОТ.
2. Якщо $t_0$, …, $t_n$ – ОТ, то $S^{n+1}\left(t_0,\ldots,t_n\right)$ – ОТ.
3. Якщо $t_0$, $t_1$ – ОТ, то $R\left(t_0,t_1\right)$ – ОТ.
4. Якщо $t$ – ОТ, то $M\left(t\right)$ – ОТ.

Завдання ЧРФ операторними термами неоднозначне, тому і така нумерація ЧРФ є неоднозначна.

Кодування ОТ задаємо так:

1) $\gamma\left(O\right) = 4$, $\gamma\left(S\right) = 8$, $\gamma\left(I_n^m\right) = 4 \cdot \left(C\left(m,n\right) - 1\right)$.

2) $\gamma\left(S^{n+1}\left(t_0,\ldots,t_n\right)\right) = 4 \cdot 2^{\gamma(t_0)} \cdot 3^{\gamma(t_1)} \cdot \ldots \cdot p_n^{\gamma(t_n)} + 1$,

де $p_n$ – $n$-е просте число.

3) $\gamma\left(R\left(t_0,t_1\right)\right) = 4 \cdot 2^{\gamma(t_0)} \cdot 3^{\gamma(t_1)} + 2$.

4) $\gamma\left(M\left(t\right)\right) = 4 \cdot 2^{\gamma(t)} + 3$.

Всі натуральні числа, які не є кодами тих чи інших ОТ, вважаємо кодом терма $O$.

## Геделівська нумерація ПРФ

Геделівська нумерація усіх ПРФ задається на основі кодування операторних термів алгебри ПРФ.

Завдання ПРФ операторними термами неоднозначне, тому и така нумерація ПРФ є неоднозначна.

Кодування ОТ задаємо так:

1) $\gamma\left(O\right) = 3$, $\gamma\left(S\right) = 6$, $\gamma\left(I_n^m\right) = 3 \cdot \left(C\left(m,n\right) - 1\right)$.

2) $\gamma\left(S^{n+1}\left(t_0,\ldots,t_n\right)\right) = 3 \cdot 2^{\gamma(t_0)} \cdot 3^{\gamma(t_1)} \cdot \ldots \cdot p_n^{\gamma(t_n)} + 1$,

де $p_n$ – $n$-е просте число.

3) $\gamma\left(R\left(t_0,t_1\right)\right) = 3 \cdot 2^{\gamma(t_0)} \cdot 3^{\gamma(t_1)} + 2$.

## Алгоритмічно нерозв'язні проблеми

За час свого існування людство придумало безліч алгоритмів для розв'язання різноманітних практичних і наукових проблем. Задаємося питанням – а чи існують які-небудь проблеми, для яких неможливо придумати алгоритми їх розв'язання?

Твердження про існування алгоритмічно нерозв'язних проблем є досить сильним – ми констатуємо, що ми не тільки зараз не знаємо відповідного алгоритму, але ми не можемо принципово ніколи його знайти.

Успіхи математики к кінцю XIX століття привели до формування думки, яку висловив Д. Гільберт – «в математиці не може бути нерозв'язних проблем», в зв'язку з цим формулювання проблем Гільбертом на конгресі 1900 року в Парижі було керівництвом до дії, констатацією відсутності розв'язків в даний момент.

Першою фундаментальною теоретичною роботою, пов'язаною з доведенням алгоритмічної нерозв'язності, була робота Курта Геделя – його відома теорема про неповноту символічних логік. Це була строго сформульована математична проблема, для якої не існує алгоритму, який її розв'язує. Зусиллями різних дослідників список алгоритмічно нерозв'язних проблем був значно розширений. Сьогодні прийнято при доведенні алгоритмічної нерозв'язності деякої задачі зводити її до задачі, що стала класичною – «***задачі зупинки***».

**Теорема.** Не існує алгоритму (машини Тьюрінга), який дозволяє за описом довільного алгоритму і його вихідних даних (і алгоритм і дані задані символами на стрічці машини Тьюрінга) визначити, чи зупиняється цей алгоритм на цих даних або працює нескінченно.

Отже, фундаментально алгоритмічна нерозв'язність пов'язана з нескінченністю виконуваних алгоритмом дій, тобто неможливістю передбачити, що для будь-яких вихідних даних розв'язок буде отримано за скінчену кількість кроків.

Проте, можна спробувати сформулювати причини, що ведуть до алгоритмічної нерозв'язності. Ці причини досить умовні, так як всі вони зводяться до проблеми зупинки, але такий підхід дозволяє більш глибоко зрозуміти природу алгоритмічної нерозв'язності.

Далі будуть сформульовані причини алгоритмічної нерозв'язності і для кожної причини наведено приклади конкретних задач.

### 1. Відсутність загального методу розв'язання задачі

Проблема 1. Розподіл дев'яток в запису числа $\pi$ .

Визначимо функцію $f(n)=i$, де $n$ – кількість дев'яток підряд в десятковому запису числа $\pi$, а $i$ – номер самої лівої дев'ятки з $n$ дев'яток підряд:

3,1415926535 8979323846 2643383279 5028841971 6939937510 5820974944 5923078164 0628620899 8628034825 3421170679 8214808651

3282306647 0938446095 5058223172 5359408128 4811174502 8410270193
8521105559 6446229489 5493038196 4428810975 6659334461 2847564823
3786783165 2712019091 4564856692 3460348610 4543266482 1339360726
0249141273 7245870066 0631558817 4881520920 9628292540 9171536436
7892590360 0113305305 4882046652 1384146951 9415116094 3305727036
5759591953 0921861173 8193261179 3105118548 0744623799 6274956735
1885752724 8912279381 8301194912 9833673362 4406566430 8602139494
6395224737 1907021798 6094370277 0539217176 2931767523 8467481846
7669405132 0005681271 4526356082 7785771342 7577896091 7363717872
1468440901 2249534301 4654958537 1050792279 6892589235 4201995611
2129021960 8640344181 5981362977 4771309960 5187072113 4999999837
2978049951 0597317328 1609631859 5024459455 3469083026 4252230825
3344685035 2619311881 7101000313 7838752886 5875332083 8142061717
7669147303 5982534904 2875546873 1159562863 8823537875 9375195778
1857780532 1712268066 1300192787 6611195909 2164201989.

Відзначимо, що $f(1) = 5$.

Задача полягає в обчисленні функції $f(n)$ для довільно заданого $n$.

Оскільки число $\pi$ є ірраціональним і трансцендентним, то ми не знаємо ніякої інформації про розподіл дев'яток (так само як і будь-яких інших цифр) в десятковому запису числа $\pi$. Обчислення значення $f(n)$ при конкретному $n$ вимагає обчислення наступних цифр в розкладанні $\pi$ доки ми не виявимо $n$ дев'яток підряд, але у нас немає загального методу обчислення $f(n)$, тому для деяких $n$ обчислення можуть продовжуватися нескінченно - ми навіть не знаємо в принципі (за природою числа $\pi$) чи існує значення $f(n)$ для всіх $n$.

Проблема 2. Обчислення досконалих чисел.

Досконалі числа – це натуральні числа, що дорівнюють сумі їх додатних дільників, не враховуючи самого числа. Наприклад: 28 = 1+2+4+7+14.

6,
28,
496,
8128,
33 550 336,
8 589 869 056,
137 438 691 328,
2 305 843 008 139 952 128,
2 658 455 991 569 831 744 654 692 615 953 842 176,
191 561 942 608 236 107 294 793 378 084 303 638 130 997 321 548 169 216,
…

Визначимо функцію $S(n) =$ $n$-е з черги досконале число і поставимо задачу обчислення $S(n)$ за довільно заданим $n$.

Немає загального методу обчислення досконалих чисел, ми навіть не знаємо, множина досконалих чисел є скінченною або зліченою, тому наш алгоритм повинен перебирати всі числа підряд, перевіряючи їх на досконалість. Відсутність загального методу розв'язання не дозволяє відповісти на питання про зупинку алгоритму.

Якщо ми перевірили $M$ чисел при пошуку $n$-ого досконалого числа – чи означає це, що його взагалі не існує?

Проблема 3. Десята проблема Гільберта.

Нехай задано многочлен $n$-ого степеня з цілими коефіцієнтами – $P_n$, чи існує алгоритм, який визначає, чи має рівняння $P_n = 0$ розв'язки в цілих числах?

Ю.В. Матіясевіч показав, що такого алгоритму не існує, тобто відсутній загальний метод визначення цілих коренів рівняння $P_n = 0$ за його цілочисельними коефіцієнтами.

## 2. Інформаційна невизначеність задачі

Проблема 4. Позиціонування машини Посту на останній позначений ящик.

Нехай на стрічці машини Посту задані набори помічених ящиків (кортежі) довільної довжини з довільними відстанями між кортежами і головка знаходиться у самого лівого поміченого ящика. Задача полягає в установці головки на самий правий позначений ящик останнього кортежу.


Спроба побудови алгоритму, що розв’язує цю задачу призводить до необхідності відповіді на питання – коли після виявлення кінця кортежу ми зсунулися вправо по порожнім ящикам на $M$ позицій і не виявили початок наступного кортежу – більше на стрічці кортежів немає або вони є десь правіше?

Інформаційна невизначеність задачі полягає у відсутності інформації або про кількість кортежів на стрічці, або про максимальну відстань між кортежами. При наявності такої інформації (при вирішенні інформаційної невизначеності) задача стає алгоритмічно розв'язною.

## 3. Логічна нерозв'язність (в сенсі теореми Геделя про неповноту)

Проблема 5. Проблема «зупинки» (див. теорему вище).

Проблема 6: Проблема еквівалентності алгоритмів.

За двома довільними заданими алгоритмами (наприклад, за двома машинами Тьюринга) визначити, чи будуть вони видавати однакові вихідні результати на будь-яких початкових даних.

Проблема 7: Проблема тотальності.

За довільним заданим алгоритмом визначити, чи буде він зупинятися на всіх можливих наборах вихідних даних. Інше формулювання цієї задачі – є чи частковий алгоритм $P$ всюди визначеним?

https://forms.office.com/Pages/ResponsePage.aspx?id=_BgIg0422UeezzxxNBjHKHI70vyivT5GiluHSZiwELdURTNENlkzUVI2OVBSNUhEQVpRQzhVMkZaNS4u