Формальні моделі алгоритмів та алгоритмічно обчислювальних функцій

МЛТА. Лекція 26.04.2021

Приклад 2. Нехай заданий алфавіт $T=\{a,b\}$ і λ – порожній символ. Перетворити слово P так, щоб на його початку виявилися всі символи a, а в кінці – усі символи b. Схема алгоритму складається з однієї підстановки:

1) $ba \to ab$

Наприклад, для слова $P = babba$ результатом буде слово $P' = aabbb$.

$P = babba$
$abbba$
$abbab$
$ababb$
$aabbb$

Приклад 3. Побудувати НА Маркова для функції усіченої різниці $f(x,y) = x \dot{-} y$.

Виберемо алфавіт $T=\{a,b,c\}$ і λ – порожній символ. Задаємо вхідне слово в вигляді $P = \underset{x \text{ разів}}{a\ldots a}\underset{y \text{ разів}}{b\ldots b}$, а вихідне слово (результат) – $P' = \underset{x \dot{-} y \text{ разів}}{c\ldots c}$. Тоді маємо:

1) $ab \to \lambda$
2) $a \to c$
3) $b \to \lambda$

Наприклад, знайти $f(2,3)=0$. Вхідне слово має вигляд $P = \underset{2}{aa}\underset{3}{bbb}$, а вихідне слово (результат) – P' – порожнє слово.

$aabbbb$
$abbb$
bb
b
λ

Наприклад, знайти $f(5,2)=3$. Вхідне слово має вигляд $P = \underbrace{aaaaa}_{5 \text{ разів}}\underset{2 \text{ рази}}{bb}$, а вихідне слово (результат) – $P' = \underset{3 \text{ рази}}{ccc}$.

$aaaaabb$
$aaaab$
aaa
caa
cca
ccc

Приклад 4. Нехай заданий алфавіт $T=\{a,b,c\}$ і λ – порожній символ. Побудуємо НА над алфавітом T, який подвоює кожен символ слова $P \in T^*$. Наприклад, вхідне слово $P = abcba$, вихідним буде слово $P' = aabbccbbaa$.

1) $*a \to aa*$
2) $*b \to bb*$
3) $*c \to cc*$
4) $* \to \cdot\lambda$
5) $a \to *a$
6) $b \to *b$
7) $c \to *c$

Продемонструємо роботу НА для деяких слів.

$P = abcba$	$P = bccba$	$P = cbcba$
$*abcba$	$*bccba$	$*cbcba$
$aa*bcba$	$bb*ccba$	$cc*bcba$
$aabb*cba$	$bbcc*cba$	$ccbb*cba$
$aabbcc*ba$	$bbcccc*ba$	$ccbbcc*ba$
$aabbccbb*a$	$bbccccbb*a$	$ccbbccbb*a$
$aabbccbbaa*$	$bbccccbbaa*$	$ccbbccbbaa*$
$aabbccbbaa$	$bbccccbbaa$	$ccbbccbbaa$

Покажемо, що алгоритм зациклюється, якщо не використовувати додатковий символ.

1) $a \to aa$
2) $b \to bb$
3) $c \to cc$

$P = abcba$	$P = bbcba$	$P = cbcba$
$aabcba$	$bbbcba$	$ccbcba$
$aaabcba$	$bbbbcba$	$cccbcba$
$aaaabcba$	$bbbbbcba$	$ccccbcba$
$\dots$	$\dots$	$\dots$

Залежно з якого символу починається слово, буде зациклювання на відповідній першому символу підстановці.

Частково-рекурсивні функції

В теорії рекурсивних функцій, як і загалом, в теорії алгоритмів, прийнятий конструктивний підхід, основною особливістю якого є те, що все множина об'єктів (тут – функцій) будується з скінченного числа початкових об'єктів (базису) за допомогою простих операцій, ефективна виконуваність яких очевидна. Операції над функціями в подальшому будемо називати операторами.

Нехай $f\left(x_1, x_2, ..., x_n\right)$ – функція, яку визначено на множині N_0^k розширеного натурального ряду $N_0 = \left\{0,1,2...\ \right\}$, тобто аргументи $x_i \in N_0$ та значення функції з N_0 (поза цієї множини функцію вважають невизначеною).

Функції такого типу називають частково-числовими функціями зчисленнозначної логіки. Позначимо за P_N^r – множину всіх частково-числових (або арифметичних) функцій.

Елементарними (***найпростішими***, ***базисними***) ***функціями*** називають такі функції:

1) $O(x)=0$ – нуль-функція;
2) $S(x)=x+1$ – функція додавання одиниці (функція слідування);
3) $I_m^n(x_1,...,x_m,...,x_n)=x_m$, $1\le m\le n$ – функція вибору аргументів (функція тотожності, координатна функція, функція введення фіктивних аргументів).

Всі елементарні функції є всюди визначеними і алгоритмічно обчислюваними функціями.

На множині P_N^r визначено три обчислювальні операції: S – суперпозиції, R – примітивної рекурсії, μ – мінімізації.

1. ***Суперпозиція***

Операція суперпозиції S^{n+1} ставить у відповідність n–арній функції $g(u_1,\dots,u_n)$ та n функціям $g_1(x_1,\dots,x_m)$, $g_2(x_1,\dots,x_m)$, …, $g_n(x_1,\dots,x_m)$ однакової арності функцію $f(x_1,\dots,x_m)$:

$$f(x_1,\dots,x_m)=g\left(g_1(x_1,\dots,x_m),\dots,g_n(x_1,\dots,x_m)\right).$$

Будемо позначати $f=S^{n+1}(g,g_1,\dots,g_n)$.

Наприклад:

$\Phi(x)=O(S(x))=0$, $\quad\Phi(x)=S(O(x))=1$, $\quad\Phi(x)=S(S(O(x)))=2$, $\Phi(x)=g(S(x),O(x))=x$, $\quad g(x,y)=x+y$.

Якщо функції g, g_1, g_2, …, g_n всюди визначені і алгоритмічно обчислювані, то функція $f=S^{n+1}(g,g_1,\dots,g_n)$ також усюди визначена і алгоритмічно обчислювана.

2. ***Операція примітивної рекурсії***

Операція примітивної рекурсії R ставить у відповідність n–арній функції g та $(n+2)$–арній функції h $(n+1)$–арную функцію f, яка задається рекурсивним визначенням:

$$\begin{cases} f(x_1,\dots,x_n,0)=g(x_1,\dots,x_n), \\ f(x_1,\dots,x_n,y+1)=h\left(x_1,\dots,x_n,y,f(x_1,\dots,x_n,y)\right). \end{cases}$$

Будем позначати $f=R(g,h)$.

Істотним в операторі примітивної рекурсії є те, що незалежно від числа змінних функції f рекурсія ведеться тільки за однією змінною y, інші n

змінних $x_1,\ldots,x_n$ на момент застосування схеми примітивної рекурсії зафіксовані і грають роль параметрів.

Це означає, що для всіх значень $x_1,\ldots,x_n,m$, значення $f\left(x_1,\ldots,x_n,m\right)$ визначається так:

$$f\left(x_1,\ldots,x_n,0\right)=g\left(x_1,\ldots,x_n\right),$$
$$f\left(x_1,\ldots,x_n,1\right)=h\left(x_1,\ldots,x_n,0,f\left(x_1,\ldots,x_n,0\right)\right),$$
$$f\left(x_1,\ldots,x_n,2\right)=h\left(x_1,\ldots,x_n,1,f\left(x_1,\ldots,x_n,1\right)\right),$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$f\left(x_1,\ldots,x_n,m\right)=h\left(x_1,\ldots,x_n,m-1,f\left(x_1,\ldots,x_n,m-1\right)\right).$$

Схема примітивної рекурсії для функції однієї змінної ($n=0$).

$$\begin{cases} f\left(0\right)=a, \\ f\left(x+1\right)=h\left(x,f\left(x\right)\right). \end{cases}$$

Якщо функції g й h всюди визначені і алгоритмічно обчислювані, то функція $f=R\left(g,h\right)$ також усюди визначена і алгоритмічно обчислювана.

3. *Операція мінімізації*

Операція мінімізації M $\left(n+1\right)-$арній функції g ставить у відповідність $n-$арну функцію f, яка задається співвідношенням:

$$f\left(x_1,\ldots,x_n\right)=\mu_y\left(g\left(x_1,\ldots,x_n,y\right)=0\right).$$

Будемо позначати $f=M\left(g\right)$.

Для всіх значень $x_1,\ldots,x_n$, значення функції $f\left(x_1,\ldots,x_n\right)$ визначається так. Послідовно обчислюємо значення $g\left(x_1,\ldots,x_n,y\right)$ для $y=0,1,2,\ldots$. Перше таке значення y, для якого $g\left(x_1,\ldots,x_n,y\right)=0$ буде шуканим значенням функції $f\left(x_1,\ldots,x_n\right)$. При цьому для всіх $t<y$ значення $g\left(x_1,\ldots,x_n,t\right)$ визначено і не дорівнює нулю.

Процес знаходження значення $f\left(x_1,\ldots,x_n\right)$ ніколи не закінчиться у таких випадках:

1) для всіх значень y значення $g\left(x_1,\ldots,x_n,y\right)$ визначено і не дорівнює нулю,
2) для всіх $t<y$ значення $g\left(x_1,\ldots,x_n,t\right)$ визначено і не дорівнює нулю, а значення $g\left(x_1,\ldots,x_n,y\right)$ невизначено,
3) значення $g\left(x_1,\ldots,x_n,0\right)$ невизначено.

Функція g може бути всюди визначеною (тотальною), а функція $f=M\left(g\right)$ навіть ніде не визначеною функцією.

Наприклад, $f\left(x\right)=\mu_y\left(x+y+1=0\right)$ – ніде не визначена функція.

$$f(x_1,x_2)=\mu_y(x_2+y=x_1)=x_1-x_2 \text{ – часткова функція.}$$

Числову функцію $f(x_1,\ldots,x_n,y)$ називають ***примітивно-рекурсивною функцією*** (скорочено ***ПРФ***), якщо вона отримана з базисних (найпростіших арифметичних) функцій $O(x)$, $S(x)$, $I_m^n(x_1,...,x_m,...,x_n), 1\le m\le n$ за допомогою скінченного числа застосувань операцій суперпозиції і примітивної рекурсії.

Функцію, яку можна отримати з базисних функцій за допомогою скінченного числа застосувань операцій суперпозиції, примітивної рекурсії і мінімізації, називають ***частково-рекурсивною функцією*** (скорочено ***ЧРФ***).

Усюди визначену ЧРФ називають ***рекурсивною функцією*** (скорочено ***РФ***).

Множину $P_{чр}$ називають класом частково-рекурсивних функцій, множину P_p – класом рекурсивних функцій, а множину $P_{пр}$ – класом примітивно-рекурсивних функцій.

З наведених визначень випливають наступні твердження:

1. Якщо функції $g_0,g_1,\ldots,g_n$ всюди визначені і алгоритмічно обчислювані, то функція $f=S^{n+1}(g_0,g_1,\ldots,g_n)$ всюди визначена і алгоритмічно обчислювана.

2. Якщо функції g і h всюди визначені і алгоритмічно обчислювані, то функція $f=R(g,h)$ всюди визначена і алгоритмічно обчислювана.

3. Якщо функція g алгоритмічно обчислювана, то і функція $f=M(g)$ алгоритмічно обчислювана.

4. Кожна ПРФ – всюди визначена, алгоритмічно обчислювана функція.

5. Каждая ЧРФ – алгоритмічно обчислювана функція.

6. Каждая РФ – всюди визначена, алгоритмічно обчислювана функція.

7. Для відповідних класів функцій мають місце співвідношення $P_{пр}\subseteq P_p\subseteq P_{чр}$.

Приклади ПРФ, ЧРФ та РФ

Наведемо приклади частково рекурсивних функцій і встановимо часткову рекурсивність основних числових функцій, використовуваних в арифметиці і аналізі.

1. Функції-константи – ПРФ.

$$f(x)=m=\underbrace{S(S(\ldots S}_{m-разів}(O(x))\ldots)\ ).$$

2. Функція $f(x_1,x_2)=x_1+x_2$ – ПРФ.

Схема примітивної рекурсії:

$$\begin{cases} f(x_1,0)=x_1+0=x_1=I_1^1(x_1), \\ f_1(x_1,x_2+1)=x_1+(x_2+1)=(x_1+x_2)+1=S(x_1+x_2)=S(f(x_1,x_2)). \end{cases}$$

Це є примітивна рекурсія за допомогою функцій $g(x_1) = I_1^1(x_1)$, $h(x_1, x_2, x_3) = I_3^3(x_1, x_2, S(x_3)) = S(x_3)$.

3. Функція $f(x_1, x_2) = x_1 \cdot x_2$ – ПРФ.

Схема примітивної рекурсії:

$$\begin{cases} f(x_1, 0) = 0 = O(x_1), \\ f_2(x_1, x_2 + 1) = x_1(x_2 + 1) = x_1 \cdot x_2 + x_1 = f(x_1, x_2) + x_1. \end{cases}$$

Це є примітивна рекурсія за допомогою функцій $g(x_1) = O(x_1)$, $h(x_1, x_2, x_3) = x_1 + x_3$. Згідно з прикладом 2 функція $h(x_1, x_2, x_3) = x_1 + x_3$ – ПРФ. Отже, f – ПРФ.

4. Функція $f(x) = sg(x) = \begin{cases} 0, \text{ якщо } x = 0 \\ 1, \text{ якщо } x > 0 \end{cases}$ – ПРФ.

Схема примітивної рекурсії:

$$\begin{cases} f(0) = 0, \\ f(x+1) = 1 = S(O(x)) \end{cases}$$

Це примітивна рекурсія, в якій $a = 0$, $h(x_1, x_2) = S(O(x_1))$.

Функція ще має позначення $sgn(x)$, $sign(x)$.

5. Функція $f(x) = nsg(x) = \begin{cases} 1, \text{ якщо } x = 0 \\ 0, \text{ якщо } x > 0 \end{cases}$ – ПРФ.

Схема примітивної рекурсії:

$$\begin{cases} f(0) = 1, \\ f(x+1) = 0 = O(x) \end{cases}$$

Це примітивна рекурсія, в якій $a = 1$, $h(x_1, x_2) = O(x_1)$.

Функція ще має позначення $\overline{sg}(x)$, $nsgn(x)$, $nsign(x)$.

6. Функція $f(x_1, x_2) = x_1 \dot{-} x_2 = \begin{cases} x_1 - x_2, & x_1 \geq x_2, \\ 0, & x_1 < x_2 \end{cases}$ (усічена різниця, будемо позначати знаком «$\dot{-}$» або «$\div$») – ПРФ.

a) Спочатку покажемо, що функція $x \dot{-} 1 = \begin{cases} x - 1, & x \geq 1, \\ 0, & x < 1, \end{cases}$ – ПРФ.

Схема примітивної рекурсії:

$$\begin{cases} 0 \dot{-} 1 = 0, \\ (x_1 + 1) \dot{-} 1 = x_1 = I_1^2(x_1, x_2). \end{cases}$$

b) Запишемо схему примітивної рекурсії для функції $x_1 \dot{-} x_2$.

$$\begin{cases} f(x_1,0) = x_1 \dot{-} 0 = x_1 = I_1^1(x_1), \\ f(x_1,x_2+1) = x_1 \dot{-} (x_2+1) = (x_1 \dot{-} x_2) \dot{-} 1 = f(x_1,x_2) \dot{-} 1. \end{cases}$$

Це примітивна рекурсія, в якій $g(x_1) = x_1$, $h(x_1,x_2,x_3) = x_3 \dot{-} 1$.

7. Функція $f(x_1,x_2) = |x_1 - x_2|$ – ПРФ.

Цю функцію можна записати в такому вигляді:

$$|x - y| = (x \dot{-} y) + (y \dot{-} x).$$

Так як для такого подання використовуються функції, примітивну рекурсивність яких доведено, то задана функція є примітивно-рекурсивною.

8. Функція $min(x_1,x_2)$ – ПРФ.

$$min(x_1,x_2) = x_1 \dot{-} (x_1 \dot{-} x_2).$$

9. Функція $max(x_1,x_2)$ – ПРФ.

$$max(x_1,x_2) = x_2 + (x_1 \dot{-} x_2).$$

10. Функція $f(x_1,x_2) = x_1 - x_2$ – ЧРФ.

Ця функція не є всюди визначеною, вона часткова. За допомогою операції мінімізації маємо

$$f(x_1,x_2) = \mu_y\left(|x_1 - (x_2 + y)| = 0\right).$$

11. Функція $f(x_1,x_2) = \left[\dfrac{x_1}{x_2}\right]$ – ціла частина від ділення x_1 на x_2. Ця функція невизначена при $x_2 = 0$, тому не є ПРФ.

Якщо довизначити цю функцію при $x_2 = 0$ так: $\left[\dfrac{x_1}{0}\right] = x_1$, тоді маємо $\left[\dfrac{x_1}{x_2}\right]$ – ПРФ.

Покажемо це. Значення n дорівнює кількості нулів в послідовності

$$1 \cdot x_2 \dot{-} x_1,\ 2 \cdot x_2 \dot{-} x_1,\ \ldots,\ n \cdot x_2 \dot{-} x_1,\ \ldots,\ x_1 \cdot x_2 \dot{-} x_1.$$

Тому маємо

$$\left[\frac{x_1}{x_2}\right] = \sum_{k=1}^{x_1} nsg(k \cdot x_2 \dot{-} x_1).$$

Наприклад, $\left[\dfrac{x_1}{x_2}\right] = \left[\dfrac{13}{3}\right] = 4$.

Маємо

$$1 \cdot 3 \dot{-} 13 = 0,\ 2 \cdot 3 \dot{-} 13 = 0,\ 3 \cdot 3 \dot{-} 13 = 0,\ 4 \cdot 3 \dot{-} 13 = 0,$$
$$5 \cdot 3 \dot{-} 13 = 2,\ 6 \cdot 3 \dot{-} 13 = 5,\ 7 \cdot 3 \dot{-} 13 = 8,\ 8 \cdot 3 \dot{-} 13 = 11,$$
$$9 \cdot 3 \dot{-} 13 = 14,\ 10 \cdot 3 \dot{-} 13 = 17, 11 \cdot 3 \dot{-} 13 = 20,\ 12 \cdot 3 \dot{-} 13 = 23.$$