## Рекурсивні функції

1. **Вступ. Базові функції.**

Першим ввів частково-рекурсивні функції Алонзо Черч. Пізніше подібні класи функцій були описані Гедельом та Кліні.

Розглядаються частково-визначені функції натурального аргументу. Кожна така функція f має – k (k>0) аргументів – натуральні числа 0,1, …, N,… і результат її обчислення – теж натуральне число - f: Nk -> N**.**Область визначення функції Df :Df ⊆ Nk. Запис, f(x1, x2, …, xk) = y, традиційно, показує що функція f застосовується до аргументів-натуральних чисел x1, x2, …, xk і її результат є натуральне число y.

*Означення 1.* Всюди визначена функція називається тотальною функцією.

Загальний метод породження (побудови) обчислювальних функцій можна описати наступним чином.

* Фіксується скінченна множина базових функцій, кожна з яких легко обчислюється.
* Фіксується скінченна множина конструктивних операцій, кожна з яких будує нову функцію з раніше побудованих обчислювальних функцій.
  + З кожною операцією зв’язується метод обчислення значення нової функції, через значення функцій з яких вона побудована.
* Інших обчислювальних функцій не існує.

*Означення 2.* Є три базові функції від одного аргументу:

z1(x1) = 0, a1(x1) = x1+ 1, (x1) = x1

котрі позначаються в виразах (операторних термах) як z1, a1, s11.

Для кожного n (n>1) є n-базових функцій селекторів, кожна з яких має n-аргументів:

(x1,x2,…,xn) = x1, (x1,x2,…,xn) = x2, …, (x1,x2,…,xn) = xn

котрі позначаються в виразах (операторних термах) як sN1, sN2, …, sNN. (В записі терма N – десяткова цифра від 1 до 9. Тобто в прикладах, що наводяться будуть використовуватися функції не більше ніж 9 аргументів. Обмеження не суттєве і використовується лише для спрощення запису.)

*Властивість 1.* Всі базові функції всюди визначені (тотальні) і обчислювані.

1. **Операція суперпозиції. Функції-константи**

*Означення 3.* Задані: функція g(x1, …, xn) від n аргументів та n функцій g1(x1, …, xm), …, gn(x1, …, xm) від m аргументів. Операція суперпозиції (g:g1,…,gn) будує з цих m+1 функцій нову функцію f(x1, …, xm) від m аргументів.

Значення функції fв точці (a1, …, am)обчислюється наступним чином:

* Вираховуються значення b1 = g1(a1, …, am), …, bn = gn(a1, …, am)функцій g1, …, gnв точці (a1, …, am).
* Вираховується значення c = g(b1, …, bn).
* Покладається f(x1, …, xm)= c.

Випадок суперпозиції, коли n=1 і m=1, в математиці називають композицією функцій. Використовуючи традиційне позначення композиції функцій, можна записати

f(x1, …, xm) = g(g1(x1, …, xm), …, gn(x1, …, xm)).

Нехай функції g, g1, …, gn позначаються (задаються) виразами rg, rg1, …, rgn , кожний з яких будується з позначень базових функцій і функцій визначених раніше. Тоді наступний вираз (rg: rg1, …, rgn) позначає функцію f. В логікі такі вирази, часто називають операторними термами.

*Властивість 2.* Якщо функції g(x1, …, xn), g1(x1, …, xm), …, gn(x1, …, xm) – обчислювані і тотальні, то функція f– обчислювана і тотальна.

*Приклад 1.* Покажемо, як використовуючи суперпозицію, будуються функції-константи.

* const0v2(x1,x2) = 0 – функція-константа 0 від двох аргументів*.* Цю функцію можна отримати як результат суперпозиції функції селектора від двох аргументів (x1,x2) в функцію z1(x1).
  + Функціюconst0v2(x1,x2) можна обчислити використовуючи формулу

const0v2(x1,x2) = z1 ((x1,x2)).

* + Вираз (операторний терм) (z1:s21) позначає цю функцію.
* const0v3(x1,x2,x3) = 0 – схожа функція, але від трьох аргументів. Отримується в результаті суперпозиції функції (x1,x2,x3) в функцію z1(x1) і позначається (z1:s31).
* const2(x1) = 2.Ця функція – результат двох суперпозицій. Спочатку функції z1(x1) в функцію a1(x1). Отримана функція a1 (z1 (x1)), її можна позначити як (a1:z1), повторно підставляється в функцію a1(x1).
  + Функція const2(x1) позначається, як (a1 : (a1:z1)), і її можна обчислити використовуючи формулу const1v2(x1,x2) = a1 (a1(z1(x1))).
* const1v2(x1,x2) = 1.Функцію можна отримати в результаті суперпозиції функції g1(x1,x2) = z1( (x1,x2)) в функцію g(x1) = a1(x1).
  + Функціюconst1v2(x1,x2)можна обчислити використовуючи формулу const1v2(x1,x2) = a1(z1( (x1,x2))) і вираз (a1 : (z1:s21)) її позначає.
* Зауважимо, що функція-константа 0 від одного аргументу – це z1.

1. **Операція примітивної рекурсії..**

*Означення 4.* Операція примітивної рекурсії [g,h] за двома функціями g(x1, …, xn) і h(x1, …, xn, xn+1, xn+2) *,* що мають, відповідно, n і (n+2) аргументи, будує нову функцію f(x1, …, xn, xn+1) , що має (n+1)*–*аргумент.

Для обчислення значення функції f в точці(a1, …, an, an+1) послідовно вирахувати наступні значення функції f:

* f(a1, …, an, 0) = g(a1, …, an)

f(a1, …, an, 1) = h(a1, …, an, 0, f(a1, …, an, 0))

*……………………………………………………….*

f(a1, …, an, an+1) = h(a1, …, an, an+1 - 1, f(a1, …, an, an+1 - 1))

Говорять, що функція f(x1, …, xn, xn+1) обраховується за наступною рекурсивною схемою (процедурою)

* f(x1, …, xn, 0) = g(x1, …, xn)
* f(x1, …, xn, y+1) = h(x1, …, xn, y, f(x1, …, xn, y))

Якщо функції g і h позначаються (задаються) виразамиrg і rh, то вираз [rg, rh ] позначає функцію f, котра отримується в результаті операції примітивної рекурсії.

*Властивість 3.* Якщо функції g(x1, …, xn) і h(x1, …, xn, xn+1, xn+2) *–* обчислювані і тотальні, то f(x1, …, xn, xn+1) *–* обчислювана і тотальна.

На практиці найчастіше операція примітивної рекурсії застосовується для визначення функцій одного аргументу f(x1) (n = 0) або двох аргументів f(x1,x2) (n = 1). В цих випадках функція *f* обраховується за схемами:

* Якщо n = 0, то g(x1) = a, функція-константа від одного аргументу, і h(x1,x2)
  + f(0 ) = a - константа
  + f(y+1) = h(y,f(y))
* Якщо n = 1, то g(x1), довільна функція від одного аргументу, і h(x1,x2,x3)
  + f(x1,0) = g(x1)
  + f(x1,y+1) = h(x1,y,f(x1,y))

1. **Прості арифметичні функції.**

Розглянемо декілька прикладів побудови найпростіших арифметичних функцій.

*Приклад 2.* notSignum(x1) = nsg(x1).

Функція nsg(x1) = 1 при x1=0 і nsg(x1) = 0 при x1 > 0. Цю функцію можна обрахувати за схемою

* nsg(0) = 1
* nsg(y+1) = 0

Якщо покласти g(x1) = 1 і h(x1,x2) =0, то функцію nsg можна отримати в результаті операції примітивної рекурсії [g,h]. Функція g(x1) – функція-константа 1 від одного аргументу, котру можна отримати в результаті суперпозиції функції z1(x1) в функцію a1(x1) (a1:z1). Функція h(x1,x2) =0 – функція-константа 0 від двох аргументів, яку раніше визначали const0v2*.* Оскільки функції g(x1) = 1 і h(x1,x2) =0, можна позначити виразами (a1:z1) і (z1:s21), то функція notSignum позначається виразом [(a1:z1),(z1:s21)].

*Приклад 3.* addition(x1,x2) = x1 +x2.

Функцію addition(x1,x2) можна обрахувати за схемою

* f(x1,0) = x1
* f(x1,y+1) = x1+y+1 = (x1+y)+1 = a1 (x1+y) = a1(f(x1,y))

Якщо покласти g(x1) = x1  і h(x1,x2,x3) = x3+1, то функцію addition можна отримати в результаті операції примітивної рекурсії [g,h]. Оскільки g(x1) – це функція-селектор, яка позначається s11, а функція h(x1,x2,x3) – результат суперпозиції функції-селектора в базову функцію a1(x1), котру можна позначити виразом (a1:s33), то функція addition позначається виразом [s11, (a1:s33)].

*Приклад 4.* multiplication(x1,x2) = x1 \*x2

Функцію можна обрахувати за схемою

* f(x1,0) = 0
* f(x1,y+1) = x1\*(y+1) = (x1\*y)+x1 = f(x1,y)+x1

Функція multiplication(x1,x2) – результат операції примітивної рекурсії до функцій g(x1) = z1(x1) і h(x1,x2,x3) = x3+ x1. Функцію можна позначити виразом

multiplication = [z1,(addition : s33,s31)] = [z1,([s11, (a1:s33)] : s33,s31)].

1. **Тотальні функції віднімання.**

*Приклад 5.* subtract1(x1) = x1÷1.

Всюди визначена функція x1÷1 = x1-1, якщо x1>0, і 0 якщо x1=0. ЇЇ можна обчислити за схемою

* f(0) = 0
* f(y+1) = (y+1) ÷1 = y

Якщо покласти g(x1) = z1(x1) і h(x1,x2) = x1, то subtract1(x1)– результат застосування операції примітивної рекурсії до константи g(x1) і функції h(x1,x2). Функцію можна позначити виразом subtract1 = [z1,s21].

*Приклад 6.* Використання попередньої функції дозволяє побудувати функцію subtraction(x1,x2) = x1÷ x2, в якої x1÷ x2= x1-x2, якщо x1≥x2, і 0 якщо x1< x2. Функцію можна обрахувати за схемою

* f(x1,0) = x1÷0 = x1
* f(x1,y+1) = x1÷ (y+1) = (x1÷y) ÷1 = f(x1,y) ÷1

Якщо покласти g(x1) = x1 і h(x1,x2,x3) = x3÷1, то subtraction(x1,x2)– результат застосування операції примітивної рекурсії до константи g(x1) і функції h(x1,x2,x3) і її можна позначити виразом

subtraction = [s11, (subtract1:s33)] = [s11, ([z1,s21]:s33)].

Використовуючи просту підстановку легко побудувати симетричну функцію subtractionRev(x1,x2) = x2÷ x1. subtractionRev(x1,x2) = subtraction( x2, x1) і позначення цієї функції subtractionRev = (subtraction : s22, s21) .

Функцію subtractionAbs(x1,x2) = |x1-x2| легко побудувати, використавши формулу subtractionAbs(x1,x2) = |x1-x2| = (x1÷ x2) + (x2÷ x1).

subtractionAbs = (addition: subtraction, subtractionRev) .

1. **Операція мінімізації.**

*Означення 5.* Операція мінімізації {g}за функцією g(x1, …, xn, xn+1) з (n+1)*-*аргументами будує функцію f(x1, …, xn) з n–аргументами, котра задається співвідношенням f(x1, …, xn)= μy (g(x1, …, xn, y) = 0) що використовує *μ*–оператор.

Обчислення *μ*–оператора - *μy* (g(x1, …, xn, y) = 0) в точці (x1, …, xn):

* Знаходить *найменше* y ≥ 0 таке, що g(x1, …, xn, z) визначена для усіх z ≤ y і g(x1, …, xn, y) = 0. Значення *μ*–оператора – *y*.
* Якщо таке *y* не можливо знайти вважається, що значення *μ*–оператора –*невизначено*.

*Означення 6.* Значення функція f ={g}в точці (a1, …, an)обчислюється наступним чином:

* Послідовно обчислюються значення g(a1, …, an,y)для y = 0, 1, 2,..
  + Результат *перше* y, коли g(a1, …, an,y)=0.
  + Для всіх t ≤ y g(a1, …, an,t) - визначено і g(a1, …, an,t) > 0.
* f (a1, …, an)- *невизначено*, якщо виконується одна з умов:
  + Для довільного y≥0 g(a1, …, an,y) - визначено і g(a1, …, an,y)>0.
  + Для довільного y < t g(a1, …, an,y) - визначено і g(a1, …, an,y)>0, але g(a1, …, an,t) - *невизначено*.
  + g(a1, …, an,0) – *невизначено.*
* Інколи обчислення кожної операції мінімізації f={g} обмежують деяким натуральним числом t, тобто обчислення виконуються лише для y = 0, 1, 2,…, t. Така операція мінімізації позначається f={g,t}

Якщо rg– вирази, що позначають функцію g, то{rg } або {rg , t} - вираз, що позначає функцію f, котра отримується в результаті операції мінімізації.

*Властивість 4.* Функція g(x1, …, xn, xn+1)може бути обчислювана і всюди визначена (тотальна), а результат операції мінімізації – функція, що обчислювана, але **не всюди** визначена.

Функція g(x1, x2) = x1+ x2 +1 *-* обчислювана і тотальна. Функція від одної змінної f = {g} *–* **всюди** невизначена функція. Функція задається співвідношенням

f(x1) = *μy* (x1+ y +1 = 0).

Для довільного y≥0 значення g(a,y)= a+y+1>0, тому для довільного a ≥ 0значення f(a) – *невизначено.*

1. **Частково визначені функції. Означення.**

*Приклад 7.* Функція subtractionAbs3(x1,x2,x3) = |x1 - (x2+x3)|отримується в результаті суперпозиції функції-селектора (x1,x2,x3) = x1 та функції g(x1,x2,x3) = addition(x2,x3) в функцію subtractionAbs

subtractionAbs3 = (subtractionAbs:s31, g) = (subtractionAbs:s31, (addition:s32,s33)).

Частково визначену функцію віднімання subtractionPart(x1,x2) = x1-x2 можна побудувати застосувавши операцію мінімізації до функції substractionAbs3.

* subtractionPart(x1,x2) = x1-x2, якщо x1 ≥ x2 і *невизначена* в інших випадках
* subtractionPart = {substractionAbs3}

Функція задається співвідношенням subtractionPart(x1,x2)= *μy* (|x1 - (x2+y)|= 0). Приклади її обчислення мають вид:

* Обчислимо subtractionPart(3,1)= 2
  + subtractionAbs3(3,1,0) = |3 - (1+0) |= 2
  + subtractionAbs3(3,1,1) = |3 - (1+1) |= 1
  + subtractionAbs3(3,1,2) = |3 - (1+2) |= 0
* Обчислимо subtractionPart(1,3)= *невизначено*
  + subtractionAbs3(1,3,0) = |1 - (3+0) |= 2
  + subtractionAbs3(1,3,1) = |1 - (3+1) |= 3
  + subtractionAbs3(1,3,2) = |1 - (3+2) |= 4
  + і так далі.

*Означення 7.* Функція, що отримується з базових функцій скінченним застосуванням операцій суперпозиції та примітивної рекурсії, називається *примітивно-рекурсивною функцією*.

Функція, що отримується з базових функцій скінченним застосуванням операцій суперпозиції, примітивної рекурсії та мінімізації, називається *частково-рекурсивною функцією.*

Всюди визначена частково-рекурсивна функція називається *рекурсивною функцією*.

*Означення 8.* Використовуючи позначення базових функцій z1, a1, sN1, sN2, …, sNN (10>N>1)та вирази, що задають (позначають) операції суперпозиції (rg : rg1, …, rgn), примітивної рекурсії [rg, rh] і мінімізації {rg } або{rg , t},кожну частково-рекурсивну функцію можна представити (задати) виразом, котрий називають *операторним термом.*

1. **Система рекурсивних функцій.**

Набір частково-рекурсивних функцій об`єднують в список означень виду

name1= expr1; name2 = expr2; …; nameN = exprN

котрий називають системою рекурсивних функції (name1,…, nameN – ідентифікатори, що задають імена функцій, а expr1,…, exprN – вирази, що задають означення цих функцій). Для скорочення і прозорості в такому списку дозволяється використовувати у виразі exprT - означення функції nameT імена функцій nameU (U<T) замісто їх означень.

Далі наводиться приклад системи рекурсивних функцій, котра включає всі функції, що наведені в тексті.

const0 = z1; const0v2 = (z1 : s21); const0v3 = (z1:s31);

const1v2 = (a1 : (z1 : s21));

const2= (a1:(a1:z1)); addition = [s11, (a1:s33)] ;

multiplication = [z1 , (addition: s33,s31)];

notSignum = [(a1:z1),(z1:s21)];

subtract1 = [z1,s21]; subtraction = [s11, (subtract1:s33)];

subtractionRev = (subtraction : s22, s21);

subtractionAbs = (addition: subtraction, subtractionRev);

subtractionAbs3=(subtractionAbs:s31, (addition:s32,s33));

subtractionPart = {subtractionAbs3, 100};