(6/27)

5 Computability

In this case: we use

- recursive functions
- while-type imperative programming language

  with $\begin{cases} \text{if} \cdots \text{then} \cdots \text{else} \cdots \\ \text{for} \\ \text{while} \end{cases}$

5.1 Primitive Recursive Function

We'll be talking about functions

$$f : \mathbb{N} \to \mathbb{N}$$

$\to$ Why $\mathbb{N}$?

$$\left\{ \text{finite words over finite alphabet } \Sigma \right\} \xrightarrow{\simeq} \mathbb{N}$$

Primitive recursive function(PR)
$\Longrightarrow$ function computable in C without while.

$$\to \begin{cases} \text{if} \\ \text{for} \end{cases} \quad \text{OK.}$$

<u>Def5.1</u>

The class of <u>PR functions</u> is inductively defined as follows.

$$\Big(\{\text{PR functions}\} \subseteq \{f : \mathbb{N}^m \to \mathbb{N} \ \ for \ \ some \ \ m\}\Big)$$

- $zero() = 0 : \mathbb{N}^0 \to \mathbb{N}$ is PR.
- $suc(x) = x + 1 : \mathbb{N} \to \mathbb{N}$ is PR.
- $proj_j^n(x_1, \cdots, x_n) = x_i : \mathbb{N}^n \to \mathbb{N}$ is PR.
- (Composition)
  $g : \mathbb{N}^m -> \mathbb{N}$ is PR.
  $g_1, g_2, \cdots, g_m : \mathbb{N}^n \to \mathbb{N}$ is PR.
  $g\Big(g_1(x_1, x_2, \cdots, x_n), g_2(x_1, x_2, \cdots, x_n), \cdots, g_m(x_1, x_2, \cdots, x_n)\Big)$ is PR.
- (Premitive Recursion)
  Let $\begin{matrix} g : \mathbb{N}^n \to \mathbb{N} \\ h : \mathbb{N}^{n+2} \to \mathbb{N} \end{matrix}$ be PR.
  Then the function $f : N^{n+1} \to \mathbb{N}$ def by
  $$\begin{cases} f(\vec{x}, 0) & = g(\vec{x}) \\ f(\vec{x}, y+1) & = h(\vec{x}, y, f(\vec{x}, y)) \end{cases}$$
  is PR.

<u>Example 5.2</u>

- Identity function
  $proj_1^1(x) = x : \mathbb{N} \to \mathbb{N}$
- factorial
  $fact(x) = x!$
  $$\begin{cases} fact(0) & = suc(zero()) \\ fact(x+1) & = suc(x) \cdot fact(x) \end{cases}$$

<u>Lem5.3</u>

If $g, g_1, g_2, \cdots, g_n$ are PR
then

$$\begin{aligned} f(x_1, x_2, \cdots, x_n) = g(&g_1(x_{i_{11}}, x_{i_{12}}, \cdots, x_{i_{1k_1}}) \\ &g_2(x_{i_{21}}, x_{i_{22}}, \cdots, x_{i_{2k_2}}) \\ &\vdots \\ &g_m(x_{i_{m1}}, x_{i_{m2}}, \cdots, x_{i_{mk_m}})) \end{aligned}$$

is PR.

Example 5.4

$add : \mathbb{N}^2 \to \mathbb{N}$ is PR

$$\begin{cases} add(x,0) & = x \\ add(x, suc(y)) & = suc(add(x,y)) \end{cases}$$

$sub : \mathbb{N}^2 \to \mathbb{N}$ is PR

$$\begin{cases} sub(x,y) & = x - y(x \geq y) \\ sub(x,y) & = 0(x < y) \end{cases} (Normalized \quad Subtraction)$$

$pred : \mathbb{N} \to \mathbb{N} : x \mapsto x - 1$ or $0(x = 0)$

$$\begin{cases} pred(0) & = zero() \\ pred(suc(x)) & = x \end{cases}$$

So,

$$\begin{cases} sub(x,0) & = x \\ sub(x, suc(y)) & = pred(sub(x,y)) \end{cases}$$

$\Big\{add, pred, sub, mult, exp, fact, max, min\Big\}$ is PR.

Lem5.5

$f : \mathbb{N}^{n+1} \to \mathbb{N}$ is PR.

Then $\left.\begin{array}{l} \sum\limits_{z<y} f(\vec{x}, z) \\ \prod\limits_{z<y} f(\vec{x}, z) \end{array}\right\} : \mathbb{N}^{n+1} \to \mathbb{N}$ are PR.

Proof

$$\begin{cases} \sum\limits_{z<0} f(\vec{x}, z) & = zero() \\ \sum\limits_{z<y+1} f(\vec{x}, z) & = add\left(\sum\limits_{z<y} f(\vec{x}, z), f(\vec{x}, y)\right) \end{cases}$$

Rem5.6

- PR functions are all total $f : \mathbb{N}^n \to \mathbb{N}$

We also talk about PR predicates.

<u>Def5.7</u>
An n-ary predicate is a subset $P \subseteq \mathbb{N}^n$.
We sometimes write $P(x_1, x_2, \cdots, x_n)$ for P.

<u>Def5.8(Characteristics function)</u>
The characteristic functions of a predicate $P \subseteq \mathbb{N}^n$ is the function.

$$\chi_P(x_1, \cdots, x_n) = \begin{cases} 0 & \text{if}(x_1, \cdots, x_n) \in P \\ 1 & \text{if}(x_1, \cdots, x_n) \notin P \end{cases}$$

<u>Def5.9</u>
A predicate $P \in \mathbb{N}^n$ is said to be premitive recursive if $\chi_P : \mathbb{N}^n \to \mathbb{N}$ is PR.

<u>Expl 5.10</u>
$\cdot$ $(\_ = 0) \subseteq \mathbb{N}$, 1-ary pred is PR.
$\because$)

$$\chi_{(\_=0)}(x) = \begin{cases} 0 & \text{if}(x = 0) \\ 1 & \text{if}(x \neq 0) \end{cases}$$

$\hookrightarrow \chi_{(\_=0)}(x) = 1 - (1 - x)$ $\qquad$ ((−) is normalized subtraction)

$\cdot$ $\lambda x \lambda y.(x = y) \subseteq \mathbb{N}^2$ is PR.
$\because$)

$$\chi_{(\_1=\_2)}(x, y) = \chi_{(\_=0)}\left((x - y) + (y - x)\right)$$

$\cdot$ $x \leq y \subseteq \mathbb{N}^2$ is PR.
$\because$)

$$\chi_{(\_1\leq\_2)}(x, y) = \chi_{(\_=0)}\left(x - y\right)$$

<u>Lem5.11</u>
Given that P, Q, R are PR.

$$\left.\begin{array}{r} \neg P, P \wedge Q, P \vee Q \\ \forall z < y.R(\vec{x}, z) \\ \exists z < y.R(\vec{x}, z) \\ R\left(f_1(\vec{x}), \cdots, f_n(\vec{x})\right) \end{array}\right\} \text{are all PR.}$$

<u>Proof</u>

$$\begin{aligned}
\chi_{\neg P} &= 1 - \chi_P \\
\chi_{P \wedge Q} &= \chi_{(_=0)}(\chi_P + \chi_Q) \\
\chi_{P \vee Q} &= \chi_P \cdot \chi_Q \\
\chi_{\forall z<y.R(\vec{x},z)} &= \chi_{(_=0)}\left(\sum_{z<y}(\vec{x},z)\right) \\
\chi_{\exists z<y.R(\vec{x},z)} &= \chi_{(_=0)}\left(\prod_{z<y}(\vec{x},z)\right)
\end{aligned}$$

<u>Lem5.12</u>

$$\begin{aligned}
g_1, \cdots, g_n : \mathbb{N}^m \to \mathbb{N}, PR \\
P_1, \cdots, P_n \subseteq \mathbb{N}^m, PR
\end{aligned}$$

Assume for each $\vec{x} \subseteq \mathbb{N}^m$,exactly one of $P_1(\vec{x}), \cdots, P_n(\vec{x})$ is true.
Then

$$f(\vec{x}) = \begin{cases} g_1(\vec{x}) & \text{if } P_1(\vec{x}) \text{ is true} \\ g_2(\vec{x}) & \text{if } P_2(\vec{x}) \text{ is true} \\ & \vdots \\ g_n(\vec{x}) & \text{if } P_n(\vec{x}) \text{ is true} \end{cases}$$

is PR.
$\because$)

$$f(\vec{x}) = \sum_{y<n} g_{y+1}(\vec{x}) \cdot \left(1 - \chi_{P_y+1}(\vec{x})\right)$$

<u>Lem5.13(Bounded least solution)</u>
$P \subseteq \mathbb{N}^{n+1}$,PR
Then

$$\mu_{z<y}P(\vec{x},y) = \begin{cases} z_0 & (\text{if} P(\vec{x},0), P(\vec{x},1), \cdots, P(\vec{x},z_0-1) \text{ are false, } P(\vec{x},z_0)\text{is true}) \\ y & (\text{if} P(\vec{x},0), P(\vec{x},1), \cdots, P(\vec{x},y-1) \text{ are false}) \end{cases}$$

is PR. <u>Proof</u>

$$\mu_{z<y}P(\vec{x},y) = \sum_{z<y} \chi_{\_=0}\left(\prod_{u<z+1} \chi_P(\vec{x},u)\right)$$

Cor5.14
$x \div y$ is PR.
$\because$)

$$x \div y = \mu_{z<y}\,(x < y \times (z+1))$$

Lem5.15
$f : \mathbb{N} \to \mathbb{N}$
$f^{\#} : \mathbb{N}^2 \to \mathbb{N}$ is defined by

$$f^{\#}(x, 0) = x$$
$$f^{\#}(x, (n+1)) = f(f^{\#}(x, n))$$

then $f^{\#}$ is PR.
Proof obvious from primitive recursion.

Lem5.16
$pr : \mathbb{N} \to \mathbb{N}$, $pr(x)$ =(the $(x+1)$ th smallest prime number) is PR.
Proof(Exercise)
Hint: $prime(x) = \begin{cases} 0 & (x \text{ is prime number}) \\ 1 & (x \text{ is not prime number}) \end{cases}$

Hint2: $pr(x+1)$ is not bigger than $pr(x)! + 1$

5.2 Recursive function

Def5.17
A partial function $f : \mathbb{N}^n \dashrightarrow \mathbb{N}$ is recursive if it is constructed in the following way.

- $0, suc, proj_j^n$ are rec.
- composition
- primitive recursion
- least solution operator : $g : \mathbb{N}^{n+1} \dashrightarrow \mathbb{N}$
  $\mu y.\Big(g(\vec{x}, y) = 0 \wedge \forall y' < y.g(\vec{x}, y') \text{ is defined}\Big) : \mathbb{N}^n \dashrightarrow \mathbb{N}$ is rec.
  (μy : the least y such that $\cdots$)

Rem5.18

In this lecture	in some literature
recursive func $N^n \dashrightarrow \mathbb{N}$	partial recursive func $N^n \dashrightarrow \mathbb{N}$
total recursive func $N^n \to \mathbb{N}$	recursive func $N^n \to \mathbb{N}$

Def5.19

A predicate $P \subseteq \mathbb{N}^n$ is recursive

$\overset{\text{def}}{\Longleftrightarrow}$ its characteristic func χ_P is $\begin{cases} \text{- total} \\ \text{- recursive} \end{cases}$

Note

this is different from

$$\chi_P(\vec{x}) = 0 \Longleftrightarrow P(\vec{x}) = true$$

it is possible that $P(\vec{x}) = false$ and $\chi_p(\vec{x}) = undefined$.

Lem5.20(Case distinction)

$$f(\vec{x}) = \begin{cases} g_1(\vec{x}) & \text{if } P_1(\vec{x}) \text{ is true} \\ & \vdots \\ g_n(\vec{x}) & \text{if } P_n(\vec{x}) \text{ is true} \end{cases}$$

where

$g_i(\vec{x})$: recursive functions

$P_i(\vec{x})$: recursive predicates

This f is recursive.