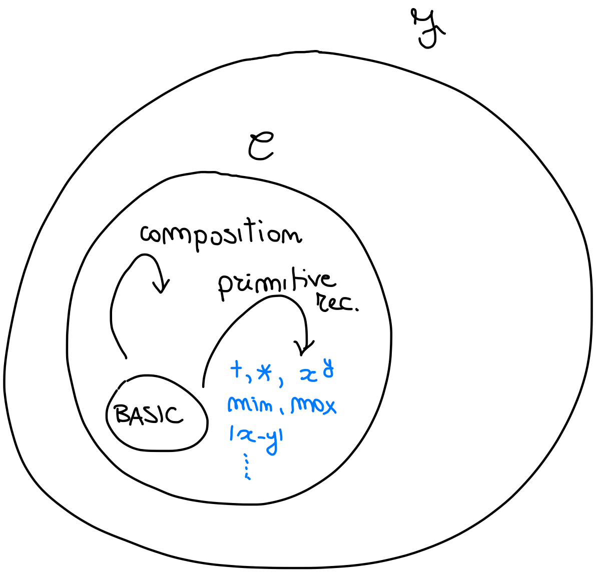
COMPUTABILITY (28/10/2024)

Class $\mathcal{C}$ of URM - computable functions



* contains the BASIC FUNCTIONS

  (a) zero
  (b) successor
  (c) projections

* closed under

  (1) (generalised) composition
  (2) primitive recursion
  (3) (unbounded) minimalisation

* OBSERVATION : Definition by cases

Let $f_1, \dots, f_m : \mathbb{N}^k \to \mathbb{N}$ functions computable total

$Q_1(\vec{x}), \dots, Q_m(\vec{x}) \subseteq \mathbb{N}^k$ predicates decidable s.t. $\forall \vec{x} \in \mathbb{N}^k \; \exists! j$ s.t. $Q_j(\vec{x})$

and we let $f : \mathbb{N}^k \to \mathbb{N}$

$$f(\vec{x}) = \begin{cases} f_1(\vec{x}) & \text{if } Q_1(\vec{x}) \\ f_2(\vec{x}) & \text{if } Q_2(\vec{x}) \\ \vdots & \vdots \\ f_m(\vec{x}) & \text{if } Q_m(\vec{x}) \end{cases}$$

Then f is computable. total

proof

computable by hyp.

$$f(\vec{x}) = \underbrace{f_1(\vec{x}) \cdot \chi_{Q_1}(\vec{x})}_{\begin{cases} f_1(\vec{x}) & \text{if } Q_1(\vec{x}) \text{ is true} \\ 0 & \text{if } Q_1(\vec{x}) \text{ is false} \end{cases}} + f_2(\vec{x}) \cdot \chi_{Q_2}(\vec{x}) + \dots + f_m(\vec{x}) \cdot \chi_{Q_m}(\vec{x})$$

$\chi_{Q_1}(\vec{x})$: $\begin{cases} 0 & \text{if } Q_1(\vec{x}) \text{ true} \\ 1 & \text{if } Q_1(\vec{x}) \text{ false} \end{cases}$

proved computable

f is the composition of computable functions hence it is computable total □

Example : $n = 2$ $\quad f_1(x) = x \;\; \forall x$, $\quad f_2(x)\uparrow \;\; \forall x$ $\qquad Q_1(x) = \text{true}$, $\quad Q_2(x) = \text{false}$

$$f(x) = \begin{cases} f_1(x) & \text{if } \overbrace{Q_1(x)}^{\text{true}} \\ f_2(x) & \text{if } \underbrace{Q_2(x)}_{\text{false}} \end{cases} = f_1(x) \qquad \forall x$$

$$\neq \underbrace{f_1(x)\cdot\chi_{Q_1}(x) + \underset{\uparrow\, \forall x}{f_2(x)\cdot\chi_{Q_2}(x)}}_{\uparrow \;\; \forall x} = f_2(x)$$

* Algebra of decidability

Let $Q(\vec{x}), Q'(\vec{x}) \subseteq \mathbb{N}^k$ be decidable predicates. Then

① $\neg Q(\vec{x})$

② $Q(\vec{x}) \wedge Q'(\vec{x})$ $\qquad$ are decidable.

③ $Q(\vec{x}) \vee Q'(\vec{x})$

proof

① $$\chi_{\neg Q}(\vec{x}) = \begin{cases} 1 & \text{if } \overbrace{\neg Q(\vec{x})}^{\chi_Q(\vec{x})=0} \\ 0 & \text{if } \underbrace{Q(\vec{x})}_{\chi_Q(\vec{x})=1} \end{cases} = \overline{sg}(\chi_Q(\vec{x})) = 1 \dot{-} \chi_Q(\vec{x})$$

↑ composition of computable functions → computable

② $$\chi_{Q\wedge Q'}(\vec{x}) = \chi_Q(\vec{x})\cdot\chi_{Q'}(\vec{x})$$

③ $$\chi_{Q\vee Q'}(\vec{x}) = sg\left(\chi_Q(\vec{x}) + \chi_{Q'}(\vec{x})\right)$$

* Bounded Sum / Product

Let $\quad f : \mathbb{N}^{k+1} \to \mathbb{N} \qquad f(\vec{x}, z)$ $\quad$ total function

define $\quad h : \mathbb{N}^{k+1} \to \mathbb{N} \qquad h(\vec{x}, y) = f(\vec{x},0) + f(\vec{x},1) + \cdots + f(\vec{x}, y-1)$

$$= \sum_{z<y} f(\vec{x}, z)$$

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

h is total and computable (by primitive recursion)

$\prod_{z<y} f(\vec{x},z)$ $\qquad$ $\prod_{z<0} f(\vec{x},z) = 1$

$$\prod_{z<y+1} f(\vec{x},z) = \left(\prod_{z<y} f(\vec{x},z)\right) \cdot f(\vec{x},y)$$

* Bounded Quantification

$Q(\vec{x},z) \subseteq \mathbb{N}^{k+1}$ decidable predicate

① $P(\vec{x},y) \equiv \forall z<y \,.\, Q(\vec{x},z)$

② $\exists z<y \,.\, Q(\vec{x},z)$

decidable $\qquad$ [Exercise]

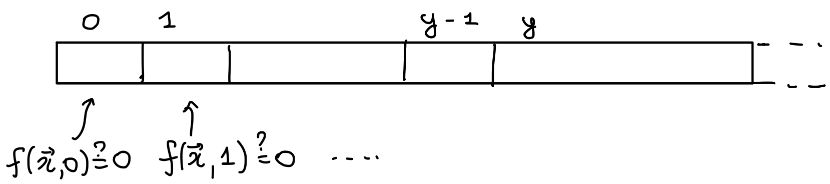
* Bounded Minimalisation

Given $f : \mathbb{N}^{k+1} \to \mathbb{N}$ total

define $h : \mathbb{N}^{k+1} \to \mathbb{N}$

$$h(\vec{x},y) = \begin{cases} \text{minimum } z<y \text{ s.t. } f(\vec{x},z)=0, & \text{if it exists} \\ y & \text{if no such } z \text{ exists} \end{cases}$$

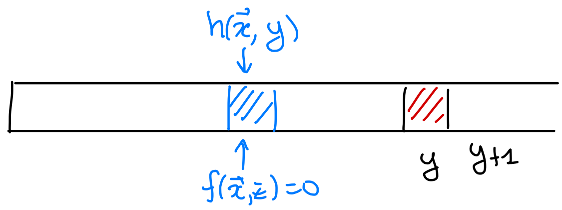
$$= \mu z<y \,.\, f(\vec{x},z) \;\cancel{(=0)}$$



Proposition: If $f : \mathbb{N}^{k+1} \to \mathbb{N}$ is total computable then $\mu z<y \,.\, f(\vec{x},z)$ is total computable

Proof

We define $h(\vec{x},y) = \mu z<y \,.\, f(\vec{x},z)$ by primitive recursion



$$\begin{cases} h(\vec{x},0) = 0 \\ h(\vec{x},y+1) = \begin{cases} \text{if } h(\vec{x},y) < y & \rightsquigarrow \boxed{h(\vec{x},y)} \\ \text{if } h(\vec{x},y) = y & \rightsquigarrow \begin{cases} \text{if } f(\vec{x},y) = 0 & \rightsquigarrow y \;\boxed{= h(\vec{x},y)} \\ \text{if } f(\vec{x},y) \neq 0 & \rightsquigarrow y+1 = h(\vec{x},y)+1 \end{cases} \end{cases} \end{cases}$$

$$= h(\vec{x},y) + \underbrace{\overline{sg}(y \dot{-} h(\vec{x},y))}_{1 \text{ iff } h(\vec{x},y)=y} \cdot \underbrace{sg(f(\vec{x},y))}_{1 \text{ iff } f(\vec{x},y)\neq 0}$$

$\Rightarrow$ h computable by primitive recursion.

$\square$

<u>OBSERVATION</u>: The following functions are computable

* $div : \mathbb{N}^2 \to \mathbb{N}$

$$div(x,y) = \begin{cases} 1 & \text{if } x \text{ divides } y \\ 0 & \text{otherwise} \end{cases} = \overline{sg}(rm(x,y))$$

* $D : \mathbb{N} \to \mathbb{N}$

$$D(x) = \text{number of divisors of } x = \sum_{y \leq x} div(y,x)$$

$\uparrow$ $y < x+1$

* $$Pr(x) = \begin{cases} 1 & x \text{ is prime} \\ 0 & \text{otherwise} \end{cases}$$

x is prime iff the only divisors of x are x and 1 & $x \neq 1$

iff x has exactly 2 divisors

$$Pr(x) = \overline{sg}(|D(x) - 2|)$$

where

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

* $p_x = x^{th}$ prime number

$p_0 = 0$ $\quad p_1 = 2$ $\quad p_2 = 3$ $\quad p_3 = 5$ $\quad p_4 = 7$ $\quad \dots$

in fact, by primitive recursion

$p_0 = 0$

P_{x+1} = " " $\mu z \leq ?$. $\underbrace{z \text{ is prime}}_{P_2(z) = 1}$ & $\underbrace{z > p_x}_{sg(z \dot{-} p_x) = 1}$ " "

$= \mu z \leq \underbrace{\left(\prod_{i=1}^{x} p_i\right)+1}$. $\overline{sg}\left(P_2(z) \cdot sg(z \dot{-} p_x)\right)$

$\uparrow$ 0 iff condition is true

let p be a prime factor

then $p \neq p_i \quad \forall i = 1 .. x$

otherwise if $p = p_J$ for some $J = 1 .. x$

then p divides $\prod_{i=1}^{x} p_i$

since p " $\left(\prod_{i=1}^{x} p_i\right)+1$

then p " 1

$\hookrightarrow p = 1$ not prime

hence $p > p_x$ i.e. $p \geq p_{x+1}$

thus $p_{x+1} \leq p \leq \left(\prod_{i=1}^{x} p_i\right)+1$

* $(x)_y$ = exponent of p_y in the prime factorisation of x

20
"
$2^2 \cdot 3^0 \cdot 5^1 \cdot 7^0 \ldots$

$(20)_1$ = exponent of $p_1 = 2$ $\rightsquigarrow (20)_1 = 2$

$(20)_2$ = " " $p_2 = 3$ $\rightarrow (20)_2 = 0$

$(20)_3$ = " " $p_3 = 5$ $\rightarrow (20)_3 = 1$

$(20)_4 = 0$

$(20)_5 = 0$

$\vdots$

$(x)_y$ = max z . p_y^z divides x

= max z . $div(p_y^z, x) = 1$

= min z . $div(p_y^{z+1}, x) = 0$

= $\mu z \leq x$. $div(p_y^{z+1}, x)$

EXERCISE: All functions which obtained from the basic functions by composition and primitive recursion are total.

* Fibonacci

$$\begin{cases} f(0) = 1 \\ f(1) = 1 \\ f(n+2) = f(n) + f(n+1) \end{cases}$$

not a primitive recursion

$g : \mathbb{N} \to \mathbb{N}^2$

$g(n) = (f(n), f(n+1))$

$D = \mathbb{N}^2$

$\pi : \mathbb{N}^2 \to \mathbb{N}$ bijective "effective"

$\pi^{-1} : \mathbb{N} \to \mathbb{N}^2$ "effective"

$\pi(x,y) := 2^x (2y+1) \dot{-} 1$ computable

$\pi^{-1} : \mathbb{N} \to \mathbb{N}^2$

$\pi^{-1}(n) = (\pi_1(n), \pi_2(n))$

$n = 2^x (2y+1) \dot{-} 1$

$n+1 = 2^x \underbrace{(2y+1)}$

$\pi_1, \pi_2 : \mathbb{N} \to \mathbb{N}$

$\pi_1(n) = (n+1)_1$

$\pi_2(n) = \left(\frac{n+1}{2^{\pi_1(n)}} \dot{-} 1\right) / 2$

π_1, π_2 computable

π^{-1} "effective"

$$\begin{cases} g : \mathbb{N} \to \mathbb{N} \\ g(n) = \pi(f(n), f(n+1)) \end{cases}$$

$g(0) = \pi(f(0), f(1)) = \pi(1,1) = 2^1(2 \cdot 1 + 1) \dot{-} 1 = 5$

$$g(n+1) = \pi\big(\underbrace{f(n+1)}_{\pi_1(g(n))},\ \underbrace{f(n+2)}_{f(n)+f(n+1) \,=\, \pi_1(g(n)) + \pi_2(g(n))}\big)$$

$$= \pi\big(\pi_1(g(n)),\ \pi_1(g(n)) + \pi_2(g(n))\big)$$

$\downarrow$

g computable

(by primitive recursion)

$\Downarrow$

$f(n) = \pi_1(g(n))$ computable by composition