COMPUTABILITY (11/11/2025)

UNIVERSAL FUNCTION

Def: Given $k \geq 1$ the universal function (for functions of arity k) is

$$\Psi_U^{(k)} : \mathbb{N}^{k+1} \to \mathbb{N}$$

$$\Psi_U^{(k)}(e, \vec{x}) = \varphi_e^{(k)}(\vec{x}) \qquad \text{(well-defined)}$$

Theorem: $\Psi_U^{(k)} : \mathbb{N}^{k+1} \to \mathbb{N}$ is computable

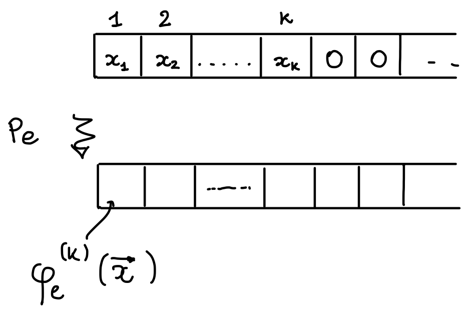
proof

given $e \in \mathbb{N}$ $\quad \vec{x} \in \mathbb{N}^k$

we want $\Psi_U^{(k)}(e, \vec{x}) = \varphi_e^{(k)}(\vec{x})$

intuitive idea $\to$ get the program $P_e = \gamma^{-1}(e)$

$\to$ execute P_e



$\to$ configuration of the memory

1	2	3		n	..	
r_1	r_2	r_3	....	r_n	0	0 - -

$$C = \prod_{i \geq 1} p_i^{r_i} \qquad r_i = (c)_i$$

$c_k : \mathbb{N}^{k+2} \to \mathbb{N}$

$c_k(e, \vec{x}, t) = \begin{cases} \text{configuration of memory after } t \text{ steps of } P_e(\vec{x}) \\ \text{(final configuration if } P_e(\vec{x}) \text{ terminates in } t \text{ or fewer steps} \end{cases}$

$J_k : \mathbb{N}^{k+2} \to \mathbb{N}$

$J_k(e, \vec{x}, t) = \begin{cases} \text{index of instruction to be executed after } t \text{ steps of } P_e(\vec{x}) \\ \text{(if } P_e(\vec{x}) \text{ does not halt in } t \text{ or fewer steps} \\ 0 \qquad \text{otherwise} \end{cases}$

$*$ ~~Obs~~ effect on configuration c of executing instruction with code i

$\rightarrow$ if $P_e(\vec{x})\downarrow$ then (c, h) is the configuration in $t_0 = \mu t. J_K(e, \vec{x}, t)$ steps

hence $\varphi_e^{(k)}(\vec{x}) = (C_K(e, \vec{x}, \mu t. J_K(e, \vec{x}, t)))_1$

$$\text{change}(c, i) = \begin{cases} \text{zero}(c, \text{Zarg}(i)) & \text{if } rm(4,i) = 0 \\ \text{succ}(c, \text{Sarg}(i)) & \text{if } rm(4,i) = 1 \\ \text{transf}(c, \text{Targ}_1(i), \text{Targ}_2(i)) & \text{if } rm(4,i) = 2 \\ \dots & \text{if } rm(4,i) = 3 \end{cases}$$

$\rightarrow$ if $P_e(\vec{x})\uparrow$ then $\mu t. J_K(e, \vec{x}, t)\uparrow$

$$\varphi_e^{(K)}(\vec{x}) = (C_K(e, \vec{x}, \mu t. J_K(e, \vec{x}, t)))_1 \uparrow$$

computable

Therefore

$$\psi_U^{(K)}(e, \vec{x}) = \varphi_e^{(K)}(\vec{x}) = (C_K(e, \vec{x}, \mu t. J_K(e, \vec{x}, t)))_1$$

if we show J_K, C_K computable we conclude.

$\boxed{\text{AIM: } C_K, J_K \text{ computable}}$

$*$ given $i \in \mathbb{N}$ instruction code i.e. $i = \beta(\text{Instr})$

$\text{Zarg}(i) = qt(4, i) + 1$	$i = \beta(Z(m)) = 4 * (m-1)$
$\text{Sarg}(i) = qt(4, i) + 1$	$i = \beta(S(m)) = 4 * (m-1) + 1$
$\text{Targ}_1(i) = \pi_1(qt(4, i)) + 1$	$i = \beta(T(m, n)) = 4 * \pi(m-1, n-1) + 2$
$\text{Targ}_2(i) = \pi_2(qt(4, i)) + 1$	

$\text{Jarg}_1, \text{Jarg}_2, \text{Jarg}_3 \; \dots$ computable

$*$ effect of executing some algebraic instruction on a configuration c

r_1	r_2	r_3		r_m

$c = p_1^{r_1} \cdot p_2^{r_2} \cdot p_3^{r_3} \cdot \ldots \; p_m^{\cancel{r_m}\, 0}$

$\text{zero}(c, m) = qt(p_m^{r_m}, c) = qt(p_m^{(c)_m}, c)$

$\text{succ}(c, m) = p_m \cdot c$

$\text{transf}(c, m, n) = p_n^{(c)_m} \cdot \text{zero}(c, n)$ computable

* effect on configuration c of executing instruction with code i

$$change(c, i) = \begin{cases} zero(c, Zarg(i)) & \text{if } rm(4, i) = 0 \\ succ(c, Sarg(i)) & \text{if } rm(4, i) = 1 \\ transf(c, Targ_1(i), Targ_2(i)) & \text{if } rm(4, i) = 2 \\ c & \text{if } rm(4, i) = 3 \end{cases}$$

computable

* configuration of registers starting from configuration c and executing instruction with index t in program P_e

$$next\,conf(e, c, t) = \begin{cases} change(c, a(e, t)) & \text{if } 1 \le t \le \ell(e) \\ c & \text{otherwise} \end{cases}$$

computable

* index of next instruction to be executed after executing $i = \beta(Instr)$ and it is at position t in the program

$$ni(c, i, t) = \begin{cases} t + 1 & \text{if } (rm(4, i) \neq 3) \text{ or } \\ & (rm(4,i) = 3) \text{ and } (c)_{Jarg_1(i)} \neq (c)_{Jarg_2(i)} \\ Jarg_3(i) & \text{otherwise} \end{cases}$$

computable

* next instruction, if we execute instruction at index t of program P_e in configuration c

$$next\,instr(e, c, t) = \begin{cases} ni(c, a(e, i), t) & \text{if } 1 \le ni(c, a(e,i), t) \le \ell(e) \\ & \text{and } 1 \le t \le \ell(e) \\ 0 & \text{otherwise} \end{cases}$$

computable

Now

$$c_k(e, \vec{x}, 0) = \prod_{i=1}^{k} p_i^{x_i}$$

1		k		
x_1	– – –	x_k	0	– – –

$$J_k(e, \vec{x}, 0) = 1$$

$$c_k(e,\vec{x},t+1) = \text{next conf}(e, c_k(e,\vec{x},t), j_k(e,\vec{x},t))$$

$$j_k(e,\vec{x},t+1) = \text{next instr}(e, c_k(e,\vec{x},t), j_k(e,\vec{x},t))$$

c_k, j_k defined by primitve recursion using computable (primitive recursive) function are computable (primitive recursive)

Hence

$$\varphi_U^{(k)}(e,\vec{x}) = \varphi_e^{(k)}(\vec{x}) = \left(c_k(e, \vec{x}, \mu t.\, j_k(e,\vec{x},t))\right)_1$$

is computable $\square$

<u>Corollary</u>: The following predicates are decidable

(a) $H_k(e,\vec{x},t) \equiv$ "$P_e(\vec{x})\downarrow$ in t steps or fewer"

(b) $S_k(e,\vec{x},y,t) \equiv$ "$P_e(\vec{x})\downarrow y$ in t steps or fewer"

<u>proof</u>

(a) $\chi_{H_k} : \mathbb{N}^{k+2} \to \mathbb{N}$

$$\chi_{H_k}(e,\vec{x},t) = \begin{cases} 1 & \text{if } H_k(e,\vec{x},t) \\ 0 & \text{otherwise} \end{cases}$$

$$= \overline{sg}(\underbrace{j_k(e,\vec{x},t)}_{\substack{0 \text{ if } P_e(\vec{x})\downarrow \text{ in } t \text{ steps} \\ \neq 0 \text{ otherwise}}})$$

computable by composition

(b)

$$\chi_{S_k}(e,\vec{x},y,t) = \underbrace{\chi_{H_k}(e,\vec{x},t)}_{\substack{1 \text{ if } P_e(\vec{x})\downarrow \text{ in } t \text{ steps} \\ 0 \text{ otherwise}}} \cdot \overline{sg}(\overbrace{|(c_k(e,\vec{x},t))_1 - y|}^{\substack{0 \text{ if output } y \\ \neq 0 \text{ otherwise}}})$$

computable by composition

$W_x^{(k)}, E_x^{(k)}, \varphi_x^{(k)}$ when $k=1$ we omit it W_x, E_x, φ_x

H_k, S_k H, S

EXERCISE Computability of the inverse

Let $f : \mathbb{N} \to \mathbb{N}$ ~~total~~ injective and computable

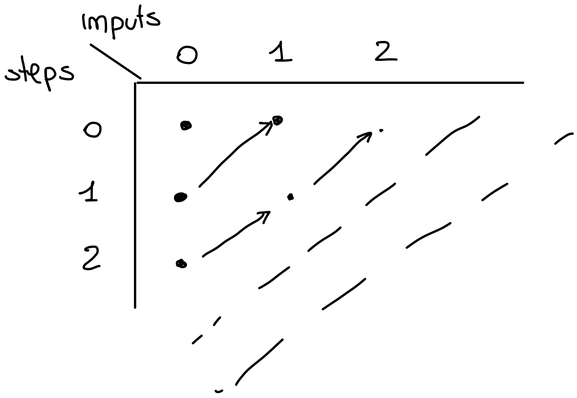
then $f^{-1} : \mathbb{N} \to \mathbb{N}$

$$f^{-1}(y) = \begin{cases} x & x \text{ such that } f(x) = y \text{, if it exits} \\ \uparrow & \text{otherwise} \end{cases}$$

is computable

$$f^{-1}(y) = \mu x. \; |f(x) - y|$$

without totality ?



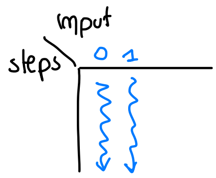
try n steps over input x

for all pairs n, x

f computable $\Rightarrow$ there is $e \in \mathbb{N}$ s.t. $f = \varphi_e$

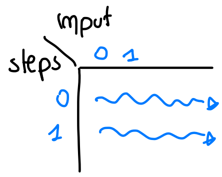
look for input x, number of steps t such that $\underbrace{\varphi_e(x) \downarrow y \text{ in } t \text{ steps}}_{S(e,x,y,t)}$

$f^{-1}(y) =$ search x, t . $S(e, x, y, t)$



$\times$ $\mu x. \, \mu t. \; S(e, x, y, t)$

$\times$ $\mu t. \, \mu x. \; S(e, x, y, t)$



$$f^{-1}(y) = \text{"}(\mu \omega . \; S(e, (\omega)_1, y, (\omega)_2))_1\text{"}$$

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

$\omega \rightsquigarrow \underbrace{(\omega)_1}_{x}, \underbrace{(\omega)_2}_{t}$

not injective

surjective !

$\omega = 3$	$2^0 \cdot 3^1 \ldots$	$\to (0, 1)$
$\omega = 6$	$2^1 \cdot 3^1 \ldots$	$\to (1, 1)$
$\omega = 30$	$2^1 \cdot 3^1 \cdot 5^1 \ldots$	$\to (1, 1)$

$$= (\mu \omega . \; |\chi_S(e, (\omega)_1, y, (\omega)_2) - 1|)_1$$

OBSERVATION : function which is total and not computable

$$f(x) = \begin{cases} \boxed{\varphi_x(x)+1} & \text{if } \varphi_x(x)\downarrow \\ 0 & \text{if } \varphi_x(x)\uparrow \end{cases}$$

computable

$$= \begin{cases} \Psi_U(x,x)+1 & \text{if } \varphi_x(x)\downarrow \\ 0 & \text{if } \varphi_x(x)\uparrow \end{cases}$$

problem ?

EXERCISE : show that the halting predicate is UNDECIDABLE

$$Halt(x) = \begin{cases} true & \text{if } \varphi_x(x)\downarrow \quad (x \in W_x) \\ false & \text{if } \varphi_x(x)\uparrow \quad (x \notin W_x) \end{cases}$$

HALTING PROBLEM

We know that a function defined by cases from total computable functions and decidable predicates is computable

If $f_1, f_2 : \mathbb{N} \to \mathbb{N}$ total computable, $Q(x)$ decidable

then
$$f(x) = \begin{cases} f_1(x) & \text{if } Q(x) \\ f_2(x) & \text{otherwise} \end{cases}$$

is computable

suggestion : remove the totality assumption.