

COMPUTABILITY (14/11/2023)

OBSERVATION: A function which is total and not computable

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

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



HALTING PROBLEM: Show that the predicate below is UNDECIDABLE

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

idea: by contradiction: we show that assuming $\text{Halt}(x)$ decidable we can prove f computable

$$f(x) = \begin{cases} \varphi_x(x) + 1 & \text{if } \varphi_x(x) \downarrow \\ 0 & \text{otherwise} \end{cases} = \begin{cases} \psi_{\sigma}(x, x) + 1 & \text{if } \text{Halt}(x) \\ 0 & \text{otherwise} \end{cases}$$

$$\cancel{\times} \quad (\psi_{\sigma}(x, x) + 1) \cdot \chi_{\text{Halt}}(x) = \begin{cases} 1 & \text{if } \text{Halt}(x) \\ 0 & \text{otherwise} \end{cases}$$

↑
when $\varphi_x(x) \uparrow$ then $\psi_{\sigma}(x, x) + 1 \uparrow \Rightarrow$ the expression is \uparrow

instead

$$\begin{aligned} f(x) &= \mu (t, y, z) \cdot \left(S(x, x, y, t) \wedge z = y + 1 \wedge \text{Halt}(x) \right) \vee \\ &\quad \left(z = 0 \wedge \neg \text{Halt}(x) \right) \\ &= \left(\mu \omega \cdot \left(S(x, x, (\omega)_2, (\omega)_1) \wedge (\omega)_3 = (\omega)_2 + 1 \wedge \text{Halt}(x) \right) \vee \right. \\ &\quad \left. \left((\omega)_3 = 0 \wedge \neg \text{Halt}(x) \right) \right)_3 \end{aligned}$$

$(\omega)_1 = t$
 $(\omega)_2 = y$
 $(\omega)_3 = z$

if you call

$$Q(x, w) \equiv (s(x, x, (w)_2, (w)_1) \wedge (w)_3 = (w)_2 + 1 \wedge \text{Halt}(x)) \vee \\ ((w)_3 = 0 \wedge \neg \text{Halt}(x))$$

decidable

$$= \left(\mu w. | \chi_Q(x, w) - 1 | \right)_3$$

computable as it arises as minimisation of computable functions

\Rightarrow contradiction

$\Rightarrow \text{Halt}(x)$ not decidable

□

EXERCISE: Let $Q(x)$ decidable predicate

$f_1, f_2 : \mathbb{N} \rightarrow \mathbb{N}$ computable

and define

$$f(x) = \begin{cases} f_1(x) & \text{if } Q(x) \\ f_2(x) & \text{if } \neg Q(x) \end{cases} \quad \text{computable}$$

proof

If f_1, f_2 total

$$f(x) = f_1(x) \cdot \chi_Q(x) + f_2(x) \cdot \chi_{\neg Q}(x)$$

if $Q(x)$	\downarrow 1	\downarrow 0
if $\neg Q(x)$	0	1

$\Rightarrow f$ computable

In general, let $e_1, e_2 \in \mathbb{N}$ st. $\varphi_{e_1} = f_1$ and $\varphi_{e_2} = f_2$

$$f(x) = \left(\mu (t, y). \left(s(e_1, x, y, t) \wedge Q(x) \right) \vee \right. \\ \left. \left(s(e_2, x, y, t) \wedge \neg Q(x) \right) \right) \rightarrow y$$

$$= \left(\mu \omega. \left(S(e_1, x, (\omega)_2, (\omega)_1) \wedge Q(x) \right) \vee \right. \\ \left. (S(e_2, x, (\omega)_2, (\omega)_1) \wedge \neg Q(x)) \right)_2$$

decidable

$\Rightarrow f$ is computable

□

EXERCISE :

$$- f(x) = \begin{cases} 0 & \varphi_x(x) \uparrow \\ 1 & \varphi_x(x) \downarrow \end{cases} \quad \text{not computable}$$

- if $\text{Halt}(x)$ is decidable then f is computable

EXERCISE : **TOTALITY**

$\text{Tot}(x) \equiv " \varphi_x \text{ is total } " \equiv " \varphi_x \text{ is terminating on every input } "$
is undecidable

In fact

$$f(x) = \begin{cases} \varphi_x(x) + 1 & \text{if } \text{Tot}(x) \\ 0 & \text{otherwise} \end{cases}$$

$$\rightarrow f \text{ is total} \quad \begin{cases} \text{Tot}(x) & \Rightarrow f(x) = \varphi_x(x) + 1 \\ \neg \text{Tot}(x) & \Rightarrow f(x) = 0 \end{cases}$$

$\rightarrow f$ is different from all total computable functions

$$(\text{if } \varphi_x \text{ is total} \Rightarrow f(x) = \varphi_x(x) + 1 \neq \varphi_x(x))$$

$\Downarrow f$ not computable

If we assume that $\text{Tot}(x)$ is decidable we derive f computable

\leadsto contradiction

In fact

$$f(x) = \begin{cases} f_1(x) & \text{if } \text{Tot}(x) \\ f_2(x) & \text{if } \neg \text{Tot}(x) \end{cases}$$

where

$$f_1, f_2 : \mathbb{N} \rightarrow \mathbb{N}$$

$$f_1(x) = \varphi_x(x) + 1 = \psi_{\varphi}(x, x) + 1 \quad \forall x$$

$$f_2(x) = 0$$

$\forall x$

computable

\Rightarrow by the previous exercise f computable, absurd.

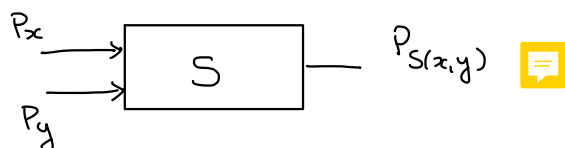
$\Rightarrow \text{Tot}(x)$ not decidable.

* EFFECTIVE OPERATIONS ON COMPUTABLE FUNCTIONS



① there exists a total computable function $S : \mathbb{N}^2 \rightarrow \mathbb{N}$

$$\forall x, y \quad \varphi_{S(x,y)}(z) = \varphi_x(z) * \varphi_y(z) \quad \forall z$$



def $P_{S(x,y)}(z)$:

$$N_x = P_x(z)$$

$$N_y = P_y(z)$$

return $N_x * N_y$

define $g : \mathbb{N}^3 \rightarrow \mathbb{N}$

$$g(x, y, z) = \varphi_x(z) * \varphi_y(z)$$

$$= \psi_{\varphi}(x, z) * \psi_{\varphi}(y, z)$$

g is computable (composition of computable functions)

Hence by (corollary of) s.m.m theorem there is $S : \mathbb{N}^2 \rightarrow \mathbb{N}$ total computable such that

$$\varphi_{S(x,y)}(z) = g(x, y, z) = \varphi_x(z) * \varphi_y(z)$$



function S takes G and x, y and "hard code" the value of x, y into G and gives back the resulting program

EXERCISE : Effectiveness of inverting a function

There is a total computable function $K: \mathbb{N} \rightarrow \mathbb{N}$

s.t. $\forall x$ if φ_x is injective then $\varphi_{K(x)} = (\varphi_x)^{-1}$



$$(\varphi_x)^{-1}(y)$$

$$z=0 \quad \varphi_x(0) \stackrel{?}{=} y$$

$$z=1 \quad \varphi_x(1) = y$$

\vdots

define

$$g(x, y) = (\varphi_x)^{-1}(y) = \begin{cases} z & \text{s.t. } \varphi_x(z) = y \quad \text{if it exists} \\ \uparrow & \text{otherwise} \end{cases}$$

(if φ_x is injective)

$$= \left(\mu(z, t) \cdot S(x, z, y, t) \right)_z$$

$$= \left(\mu \omega \cdot S(x, (\omega)_1, y, (\omega)_2) \right)_1$$

$$= \left(\mu \omega \cdot (\chi_S(x, (\omega)_1, y, (\omega)_2) - 1) \right)_1$$

computable

Hence, by smm theorem, there is $K: \mathbb{N} \rightarrow \mathbb{N}$ total computable

s.t.

$$\varphi_{K(x)}(y) = g(x, y) = (\varphi_x)^{-1}(y) \quad \text{if } \varphi_x \text{ injective}$$

What do we get when φ_x is not injective?

$\varphi_{K(x)}(y)$ is one of the counter images of y

QUESTION: Given $f: \mathbb{N} \rightarrow \mathbb{N}$ computable. Define

$$g(y) = \begin{cases} \min \{x \mid f(x) = y\} & \text{if } \exists x. \text{ s.t. } f(x) = y \\ \uparrow & \text{otherwise} \end{cases}$$

Is g computable?

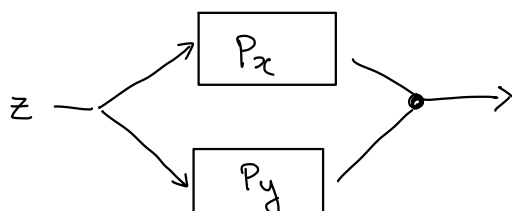
EXERCISE

There is a total computable function $s: \mathbb{N}^2 \rightarrow \mathbb{N}$

such that

$$W_{s(x,y)} = W_x \cup W_y$$

$$\varphi_{s(x,y)}(z) \downarrow \text{ iff } \varphi_x(z) \downarrow \text{ or } \varphi_y(z) \downarrow$$



$$g: \mathbb{N}^3 \rightarrow \mathbb{N}$$

$$g(x, y, z) = \begin{cases} \downarrow & \text{if } \varphi_x(z) \downarrow \text{ or } \varphi_y(z) \downarrow \\ \uparrow & \text{otherwise} \end{cases}$$

$$= \mathbb{1}(\mu t. H(x, z, t) \vee H(y, z, t))$$

where $\mathbb{1}(x) = 1 \quad \forall x$

g is computable and thus by smn theorem $\exists s: \mathbb{N}^2 \rightarrow \mathbb{N}$ total computable

s.t.

$$\varphi_{s(x,y)}(z) = g(x, y, z) = \begin{cases} 1 & \text{if } \varphi_x(z) \downarrow \text{ or } \varphi_y(z) \downarrow \\ \uparrow & \text{otherwise} \end{cases}$$

s is the desired function

$$z \in W_{s(x,y)} \text{ iff } \underbrace{\varphi_{s(x,y)}(z) \downarrow}_{\varphi(x,y,z)} \text{ iff } \varphi_x(z) \downarrow \text{ or } \varphi_y(z) \downarrow$$

$$\text{iff } z \in W_x \text{ or } z \in W_y$$

$$\text{iff } z \in W_x \cup W_y$$

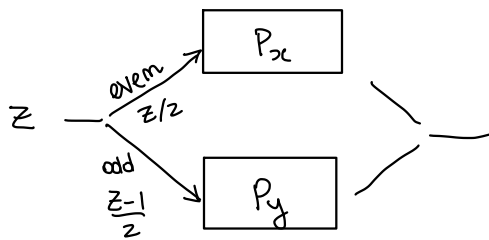
EXERCISE

There is a total computable function $s: \mathbb{N}^2 \rightarrow \mathbb{N}$

such that

$$E_{s(x,y)} = E_x \cup E_y$$

($P_{s(x,y)}$ produces as outputs all values produced by P_x and P_y)



	0	1	2	3	...
P_x	0	3	4	1	...
P_y	1	5	1	2	

$$g(x, y, z) = \begin{cases} \varphi_x(z/2) & z \text{ even} \\ \varphi_y(\frac{z-1}{2}) & z \text{ odd} \end{cases}$$

~~$$= \psi_v(x, qt(z, z)) * \overline{\psi_g(\text{em}(z, z))} + \psi_v(y, qt(z, z)) * \text{em}(z, z)$$~~

$$= \left(\mu(n, t) \cdot (S(x, z/2, n, t) \wedge z \text{ even}) \vee (S(y, \frac{z-1}{2}, n, t) \wedge z \text{ odd}) \right)_n$$

$$= \left(\mu \omega. \underbrace{(S(x, qt(z, z), (\omega)_1, (\omega)_2) \wedge z \text{ even}) \vee (S(y, qt(z, z), (\omega)_1, (\omega)_2) \wedge z \text{ odd})}_{\text{decidable}} \right)_1$$

computable

By smm theorem $\exists s: \mathbb{N}^2 \rightarrow \mathbb{N}$ total computable s.t.

$$\varphi_{s(x,y)}(z) = g(x, y, z) = \begin{cases} \varphi_x(z/2) & z \text{ even} \\ \varphi_y(\frac{z-1}{2}) & z \text{ odd} \end{cases}$$

I claim that s is the desired function, i.e. $E_{s(x,y)} = E_x \cup E_y$

$$(\leq) \quad \sigma \in E_{S(x,y)}$$

$$\exists z. \quad \text{s.t.} \quad \varphi_{S(x,y)}(z) = \top$$

hence two possibilities

$$\left. \begin{array}{l} - \nu = \varphi_x \left(\frac{z}{2} \right) \sim \nu \in E_x \\ - \nu = \varphi_y \left(\frac{z-1}{2} \right) \sim \nu \in E_y \end{array} \right\} \rightarrow \nu \in E_x \cup E_y$$

$$(\supset) \quad N \in E_x \cup E_y \quad \rightsquigarrow \quad N \in E_{S(x,y)}$$

i.e. $\textcircled{1} \quad v \in E_x \rightsquigarrow v \in E_{S(x,y)}$

$$\textcircled{2} \quad N \in E_y \quad \rightsquigarrow \quad N \in E_{S(x,y)}$$

① $N \in E_x$ i.e. $\exists z$ s.t. $\varphi_x(z) = N$

therefore $\varphi_{S(x,y)}(z) = \varphi_x\left(\frac{z}{2}\right) = \varphi_x(z) = 0$

$$\leadsto \delta \in E_S(x, y)$$

② identical

* EXERCISE : variant of URM machine

URM^P

$Z(m)$		
$S(m)$	$P(m)$	$\tau_m \leftarrow \tau_m - 1$
$T(m, m)$		
$J(m, m, t)$		

\mathcal{C}_P URM^P-computable function

$$\mathcal{C}_p \stackrel{?}{=} \mathcal{C}$$

* EXERCISE : Are there $f, g : \mathbb{N} \rightarrow \mathbb{N}$ functions s.t.

① f computable, g not computable $f \circ g$ computable

② f not computable, g not computable, $f \circ g$ computable