COMPUTABILITY (18/11/2025)

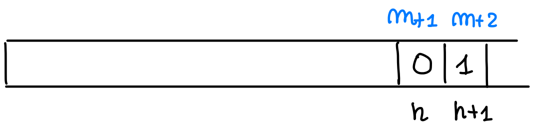
EXERCISE : URM^P instructions

$Z(m)$
$T(m,n)$
$J(m,n,t)$
$\cancel{S(m)}$ $\qquad P(m) \qquad r_m \leftarrow r_m \dot{-} 1 = \begin{cases} 0 & \text{if } r_m = 0 \\ r_m - 1 & \text{if } r_m > 0 \end{cases}$

$$\mathcal{C}^P \subsetneq \mathcal{C}$$

$(\mathcal{C}^P \subseteq \mathcal{C})$ given a program P in URM^P

$\rho(P)+1$
$\shortparallel$



$- - -$
t: $P(m)$ $\qquad J(1,1,\text{SUB})$
$- - -$

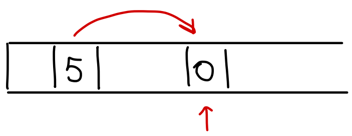
SUB : $J(m, m+1, \text{END})$
$S(m+2)$
LOOP : $J(m, m+2, \text{RES})$
$S(m+1)$
$S(m+2)$
$J(1,1,\text{LOOP})$
RES : $T(m+1, m)$
END : $J(1,1,t+1)$

FORMAL PROOF :

For every program P $\underbrace{\text{of } URM^P}_{\text{using URM instructions plus } P(m)}$, for every $K \in \mathbb{N}$, there is a URM program P' such that $f_P^{(K)} = f_{P'}^{(K)}$

(proof by induction on the number of $P(\,)$ instructions.)

$(\mathcal{C} \nsubseteq \mathcal{C}^P)$



Given a program P of URM^P the maximum value in registers after any number of steps of computation of $P(\vec{x})$ is $\leq \max_{1 \leq i \leq k} x_i$

proof by induction on the number of steps t of $P(\vec{x})$

$(t=0)$ the registers are

1		k			
x_1	...	x_k	0	0	...

$$\max_i r_i = \max_{1 \le i \le k} x_i$$

$(t \to t+1)$ the content of memory after $t+1$ steps

x_1	...	x_k	0	...

t steps $\rightsquigarrow$

r_1	r_2	r_3	...

by inductive hyp.

$$\max_i r_i \le \max_{1 \le i \le k} x_i$$

1 step $\rightsquigarrow$

r_1'	r_2'	r_3'	...

different cases according to the last instruction executed at step $t+1$

$Z(n)$
$T(m,n)$
$J(m,n,t)$
$P(n)$

$$\rightsquigarrow \quad \max_i r_i' \le \max_i r_i \le \max_{1 \le i \le k} x_i$$

$\rightsquigarrow$

The successor function $s: \mathbb{N} \to \mathbb{N}$ $\quad s(x) = x+1$ is not URM^P-computable

In fact $\quad s(0) = 1$

0	0	...

$\max = 0$

$\rightsquigarrow$

1	...

$\wedge$

$\max \ge 1$

impossible

Hence $\quad \underset{\not\ni s}{\mathcal{C}^P} \subsetneq \underset{\ni s}{\mathcal{C}}$

EXERCISE : Termination is decidable for the URM^P machine

(finite state)

EXERCISE: Show that there is a total computable function $k : \mathbb{N} \to \mathbb{N}$ such that $E_{k(x)} = W_x$

$P_x \rightsquigarrow P_{k(x)}$ set of outputs which coincides with the set of inputs where P_x terminates

$$\boxed{\begin{array}{l} \text{def } P_{k(x)}(y) \\ \quad P_x(y) \\ \quad \text{return } y \end{array}}$$

define $f : \mathbb{N}^2 \to \mathbb{N}$

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

$$= \underbrace{\underbrace{\mathbb{1}(\varphi_x(y))}_{\substack{1 \text{ if } \varphi_x(y)\downarrow \\ \uparrow \text{ otherwise}}} * y}_{\substack{y \text{ if } \varphi_x(y)\downarrow \\ \uparrow \text{ otherwise}}} = \mathbb{1}(\varphi_U(x, y)) * y$$

computable (composition of computable functions)

Hence by smn theorem there $k : \mathbb{N} \to \mathbb{N}$ total computable s.t. $\forall x, y$

$$\varphi_{k(x)}(y) = f(x, y) = \begin{cases} y & \text{if } \varphi_x(y)\downarrow \\ \uparrow & \text{otherwise} \end{cases}$$

$*$ k is the desired function : $W_x = E_{k(x)}$

($W_x \subseteq E_{k(x)}$) let $y \in W_x$ then $\varphi_x(y)\downarrow$

therefore $\varphi_{k(x)}(y) = y$. Thus $y \in E_{k(x)}$

($E_{k(x)} \subseteq W_x$) let $y \in E_{k(x)}$. Then there is $z \in \mathbb{N}$ s.t. $\varphi_{k(x)}(z) = y$

Since $\varphi_{k(x)}$ is the identity when it is defined then $z = y$ and $\varphi_x(y)\downarrow$ and thus $y \in W_x$

$\square$

EXERCISE : there is a total computable function $k : \mathbb{N} \to \mathbb{N}$ such that

$$W_{k(x)} = \mathbb{P} \qquad \text{(even numbers)}$$

$$E_{k(x)} = \{ n \in \mathbb{N} \mid n \geq x \}$$

define $f : \mathbb{N}^2 \to \mathbb{N}$

$$f(x, y) = \begin{cases} x + y/2 & \text{if } y \text{ is even} \\ \uparrow & \text{if } y \text{ is not even} \end{cases}$$

$$= x + qt(2, y) + \underbrace{\mu w .\ rm(2, y)}_{\substack{0 \text{ when } y \text{ is even} \\ \uparrow \text{ '' } y \text{ is odd}}}$$

computable.

Hence by smn theorem there is $k : \mathbb{N} \to \mathbb{N}$ total and computable such that

$$\varphi_{k(x)}(y) = f(x, y) = \begin{cases} x + y/2 & \text{if } y \text{ is even} \\ \uparrow & \text{if } y \text{ is not even} \end{cases}$$

k is the desired function

* $W_{k(x)} = \mathbb{P}$ ok

* $E_{k(x)} = \{ n \mid n \geq x \}$

$$E_{k(x)} = \{ \varphi_{k(x)}(y) \mid y \in W_{k(x)} \}$$

$$= \{ x + y/2 \mid y \in \mathbb{P} \}$$

$$= \{ x + \cancel{2}z/\cancel{2} \mid z \in \mathbb{N} \}$$

$$= \{ x + z \mid z \in \mathbb{N} \}$$

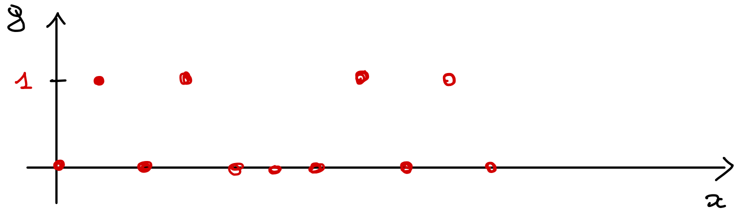
$$= \{ n \mid n \geq x \}$$

EXERCISE: Are there f, g functions s.t. f computable, g not computable

such that $f \circ g$ computable?

$$g(x) = \begin{cases} 1 & \text{if } Halt(x) \\ 0 & \text{otherwise} \end{cases}$$

not computable



$f(x) = 0 \quad \forall x$

computable

then $f \circ g\,(x) = f(g(x)) = 0$ computable

* Are there f, g functions s.t. f not computable, g not computable

such that $f \circ g$ computable?

yes:

g as before $\qquad g(x) = \begin{cases} 1 & Halt(x) \\ 0 & \text{otherwise} \end{cases}$

non computable

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

<u>non computable</u>: $\forall x > 1 \quad f \neq \varphi_x$

$\forall y \in \mathbb{N} \quad \exists x > 1$ s.t. $f \neq \varphi_x = \varphi_y \Rightarrow f \neq \varphi_y$

but $f \circ g\,(x) = f(g(x)) = 0 \quad \forall x$ computable

EXERCISE: Can every function f which is computable

be obtained as $g \circ h$ with g, h non computable?

<u>EXERCISE</u>: Prove that $pow_2 : \mathbb{N} \to \mathbb{N}$ is PR

$$pow_2(x) = 2^x$$

by using only the definition of PR.

(PR least class of functions containing basic functions and closed under composition and primitive recursion)

$\uparrow$
zero
succ
projections

$x+y$ $\begin{cases} x+0 = x \\ x+(y+1) = (x+y)+1 \end{cases}$

$x*y$ $\begin{cases} x*0 = 0 \\ x*(y+1) = (x*y)+x \end{cases}$

x^y $\begin{cases} x^0 = 1 = S(0) \\ x^{y+1} = x^y * x \end{cases}$

finally

$$pow_2(x) = 2^x = S(S(0))^x$$

alternatively

$$\begin{cases} pow_2(0) = 2^0 = 1 = S(0) \\ pow_2(y+1) = 2^{y+1} = 2^y \cdot 2 = 2 \cdot pow_2(y) = pow_2(y) + pow_2(y) \end{cases}$$

$x+y$ $\begin{cases} x+0 = x \\ x+(y+1) = (x+y)+1 \end{cases}$

hence I conclude

alternatively

$$\begin{cases} pow_2(0) = 2^0 = 1 = S(0) \\ pow_2(y+1) = 2^{y+1} = 2^y \cdot 2 = twice(pow_2(y)) \end{cases}$$

$$\begin{cases} twice(0) = 0 \\ twice(y+1) = 2(y+1) = 2y+2 = S(S(twice(y))) \end{cases}$$

EXERCISE: $\chi_{\mathbb{P}} \in PR$ ($\mathbb{P}$ = even numbers)

$$\chi_{\mathbb{P}}(x) = \begin{cases} 1 & \text{if } x \text{ is even} \\ 0 & \text{otherwise} \end{cases}$$

1^{st} approach:

$$\chi_{\mathbb{P}}(x) = \overline{sg}\,(rm(2,x))$$

$rm(\dots)$

long!

direct approach

$$\begin{cases} \chi_{\mathbb{P}}(0) = 1 = S(0) \\ \chi_{\mathbb{P}}(y+1) = \overline{sg}\,(\chi_{\mathbb{P}}(y)) \end{cases}$$

$$\begin{cases} \overline{sg}\,(0) = 1 = S(0) \\ \overline{sg}\,(y+1) = 0 \end{cases}$$