COMPUTABILITY (12/10/2026)

EXERCISE: URM^S machine : variant of URM

$T(m, n)$ ~~$T(m, n)$~~ $\quad T_S(m, n)$ $\quad r_m \longleftrightarrow r_m$

$\mathcal{C}^S \stackrel{?}{=} \mathcal{C}$

proof

$(\mathcal{C} \subseteq \mathcal{C}^S)$ By a previous exercise $\mathcal{C} = \mathcal{C}^- \subseteq \mathcal{C}^S$.

Explicitly, let $f \in \mathcal{C}$ $\quad f: \mathbb{N}^k \to \mathbb{N}$ $\quad$ we want ~~$\quad f \in \mathcal{C}^S$~~

let P be a URM-program such that $f = f_P^{(k)}$.

by a previous exercise $(\mathcal{C}^- = \mathcal{C})$ there is P' URM-program without transfer instructions such that $f_{P'}^{(k)} = f_P^{(k)}$

but P' is also a URM^S program and therefore $f = f_P^{(k)} = f_{P'}^{(k)} \in \mathcal{C}^S$

$(\mathcal{C}^S \subseteq \mathcal{C})$ Take $f \in \mathcal{C}^S$ $\quad f: \mathbb{N}^k \to \mathbb{N}$ and let P a URM^S program such that $f = f_P^{(k)}$. We want to "transform" P into a URM program say P', such that

$$f_P^{(k)} = f_{P'}^{(k)}$$

$T_S(m, n)$ $\qquad$ $T(m, i)$ $\qquad$ R_i not used by P

$T(m, m)$

$T(i, m)$

A URM^S-program P can be transformed into P' URM-program such that $f_P^{(k)} = f_{P'}^{(k)}$

We proceed by induction on h = number of $T_S(-,-)$ instructions in P

$(h = 0)$ P is already a URM-program, hence $P' = P$

$(h \to h+1)$ Let P URM^s program with $h+1$ Ts instructions

$$P \begin{cases} I_1 \\ \vdots \\ I_t \quad Ts(m,m) \\ \vdots \\ I_s \end{cases} \rightsquigarrow P'' \begin{cases} I_1 \\ \vdots \\ I_t \quad J(1,1,sub) \\ \\ I_s \\ I_{s+1} \quad J(1,1,END) \\ sub: \; T(m,i) \\ \quad T(m,m) \\ \quad T(i,m) \\ \quad J(1,1,t+1) \end{cases}$$

assuming:

- P well-formed (if it halts, it does at $s+1$)
- register i not used by P

$$i = \max\left(\{ j \mid R_j \text{ is used by } P\} \cup \{k\}\right) + 1$$

Then P'' is such that

$$f_{P''}^{(k)} = f_P^{(k)} \quad (*)$$

and it contains h swap instructions. Hence by inductive hyp. there is P' URM-program such that $f_{P'}^{(k)} = f_{P''}^{(k)}$ $(*)$

Summing up

$$f = f_P^{(k)} \underset{(*)}{=} f_{P''}^{(k)} \underset{(*)}{=} f_{P'}^{(k)}$$

and thus $f \in \mathcal{C}$.

The proof is wrong: I am using the inductive hyp. on P'' which is not a URM^s program (it uses both T and Ts)

Solution: prove a stronger assertion

> Every program P which uses both swap Ts and transfers T can be transformed into a URM-program P' s.t. $f_P^{(k)} = f_{P'}^{(k)}$

□

EXERCISE : Consider URM$^=$ without jump instructions

Show (a) $\mathcal{E}^= \subsetneq \mathcal{E}$ and (b) characterise the shape of functions in $\mathcal{E}^=$

proof

(a) An URM$^=$ program

$P \begin{cases} I_1 \\ \vdots \\ I_s \end{cases}$ $\ell(P) = s$ length of program P

P terminates after $\ell(P)$ steps

$\rightsquigarrow$ all functions in $\mathcal{E}^=$ are total $\rightsquigarrow \mathcal{E}^= \subsetneq \mathcal{E}$

e.g. $f : \mathbb{N} \to \mathbb{N}$ $f \in \mathcal{E}$ $J(1,1,1)$

$f(x) \uparrow \quad \forall x \in \mathbb{N}$ $f \notin \mathcal{E}^=$ because is not total

(saying "the program uses jumps" is not sufficient for $f \notin \mathcal{E}^=$

e.g. $I_1 : J(1,1,2)$ P

$f_P^{(1)}(x) = x$

but $f_P^{(1)} \in \mathcal{E}^=$

(b)

shape of functions in $\mathcal{E}^=$?

1
x | 0 | 0 | - - -

conjecture :

$f(x) = c$ (constants)

or $f(x) = x + c$ (increment by a constant)

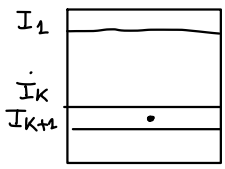
$c \in \mathbb{N}$ suitable constant

denote $r_1(x, K)$ = content of register 1 after K steps starting from $x|0|0|---$

We prove by induction on K that $r_1(x,K) \begin{cases} c \\ x+c \end{cases}$ $c \in \mathbb{N}$

$(k=0)$ $\quad r_1(x,0) = x = x+0 \quad c=0$

$(k \to k+1)$ By induction hyp $\quad r_1(x,k) = \begin{cases} c \\ x+c \end{cases}$



different cases according to the shape of I_{k+1}

* I_{k+1} $\quad Z(m)$ $\quad$ two sub cases

  $(m=1) \quad r_1(x,k+1) = 0$ ok.

  $(m>1) \quad r_1(x,k+1) = r_1(x,k)$ ok by ind. hyp.

* I_{k+1} $\quad S(m)$ $\quad$ two subcases

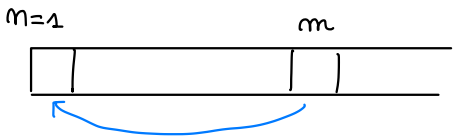
  $(m=1) \quad r_1(x,k+1) = \overbrace{r_1(x,k)}^{\substack{c \\ x+c}} + 1 \quad$ ind. hyp.

  $\begin{cases} c+1 \\ x+(c+1) \end{cases}$ ok!

  $(m>1) \quad r_1(x,k+1) = r_1(x,k)$ ok by ind. hyp.

* I_{k+1} $\quad T(m,n)$ $\quad (n>1$ or $m=1) \quad r_1(x,k+1) = r_1(x,k)$ ok, by ind hyp.

$(m=1$ and $n>1)$



not working, no info on $R_m \quad m>1$ ...

The key observation is that the same property holds for all registers

$r_j(x,k)$ = content of R_j after k steps of computation starting from $\boxed{x|0|0|--}$

show by induction that $\forall k \ \forall j$ (induction on k)

$$r_j(x,k) = \begin{cases} c \\ x+c \end{cases}$$

The proof now works smoothly. [EXERCISE]

EXERCISE

for n-ary functions

$$f(x_1, \dots, x_n) = \begin{cases} c \\ x_j + c \end{cases} \qquad 1 \le j \le n, \quad c \in \mathbb{N} \text{ constant}$$

* Decidable predicates

$div(x, y)$ ≡ "x divides y"

$div \subseteq \mathbb{N} \times \mathbb{N}$

$div = \{ (m, m*k) \mid m, k \in \mathbb{N} \}$

or

$div : \mathbb{N} \times \mathbb{N} \to \{ true, false \}$

$\{ 1, 0 \}$

K-ary predicate $\quad Q(x_1, \dots, x_k) \subseteq \mathbb{N}^k$

$Q : \mathbb{N}^k \to \{0, 1\}$

$\{0, 1\}$: ↑ false, ↑ true

Def. (decidable predicate)

Let $Q(x_1, \dots, x_k) \subseteq \mathbb{N}^k$ a predicate. It is decidable if

$$\chi_Q : \mathbb{N}^k \to \mathbb{N}$$

$$\chi_Q(x_1, \dots, x_k) = \begin{cases} 1 & \text{if } Q(x_1, \dots, x_k) \text{ holds} \\ 0 & \text{otherwise} \end{cases}$$

is URM-computable

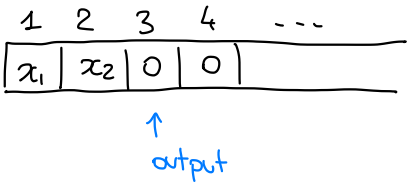
Example: $\quad Q(x_1, x_2) \subseteq \mathbb{N}^2$

$Q(x_1, x_2) =$ "$x_1 = x_2$" $\quad$ decidable

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

$$\chi_Q(x_1, x_2) = \begin{cases} 1 & \text{if } x_1 = x_2 \\ 0 & \text{otherwise} \end{cases}$$

computable



J(1,2, YES)

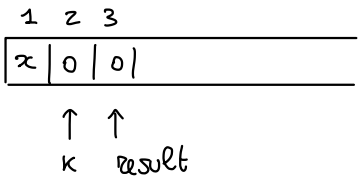
NO : J(1,1, RES)

YES : S(3)

RES : T(3,1)

Example : Q(x) ≡ "x evem" decidable EXERCISE

SOLUTION



EVEN : J(1,2, TRUE)
S(2)

ODD : J(1,2, FALSE)
S(2)
J(1,1, EVEN)

TRUE : S(3)

FALSE : T(3,1)

* **Computability on other domains**

D countable

$\alpha : D \to \mathbb{N}$ bijective "effective"

(inverse α^{-1} effective)

A^*, $\mathbb{Q}$, $\mathbb{Z}$, ... ~~$\mathbb{R}$~~

Given $f : D \to D$ function is computable if

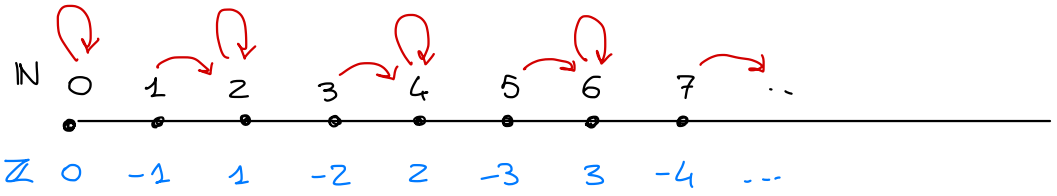
$$\begin{array}{ccc} D & \xrightarrow{f} & D \\ {\scriptstyle \alpha^{-1}}\uparrow & & \downarrow{\scriptstyle \alpha} \\ \mathbb{N} & \xrightarrow{f^*} & \mathbb{N} \end{array}$$

$f^* = \alpha \circ f \circ \alpha^{-1} : \mathbb{N} \to \mathbb{N}$ is URM-computable

Example: Computability on $\mathbb{Z}$

$\alpha : \mathbb{Z} \to \mathbb{N}$

$$\alpha(z) = \begin{cases} 2z & \text{if } z \geq 0 \\ -2z-1 & \text{if } z < 0 \end{cases}$$



$\alpha^{-1} : \mathbb{N} \to \mathbb{Z}$

$$\alpha^{-1}(m) = \begin{cases} m/2 & m \text{ even} \\ -\frac{m+1}{2} & m \text{ odd} \end{cases}$$

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

$f(z) = |z|$ computable

$f^* : \alpha \circ f \circ \alpha^{-1} : \mathbb{N} \to \mathbb{N}$

$$f^*(n) = \alpha f \alpha^{-1}(n) = \begin{cases} n \text{ even} & \alpha \overbrace{f\left(\frac{n}{2}\right)} = \alpha\left(\frac{n}{2}\right) = 2\frac{n}{2} = n \\ n \text{ odd} & \alpha f\left(-\frac{n+1}{2}\right) = \alpha\left(\frac{n+1}{2}\right) = 2\frac{n+1}{2} = n+1 \end{cases}$$

$$= \begin{cases} n & n \text{ even} \\ n+1 & n \text{ odd} \end{cases}$$

URM- computable

$\Rightarrow$ f is computable