

PSET 4 — 04/26/2024

Prof. Miller

Student: Amittai Siavava

Problem 2.

Recall that W_e is $\text{dom}(\varphi_e)$, and that X is c.e. if $X = W_e$ for some e . Show that it is equivalent to define the c.e. sets as those that are either finite or the range of a total, computable, injective function $f : \mathbb{N} \rightarrow \mathbb{N}$.

We can show the equivalence of the two definitions by proving that X is c.e. if and only if X is finite or the range of a total, computable, injective function.

\implies

Proof. Let X be c.e. so that $X = W_e$ for some e .

If X is finite, then the condition is satisfied.

Suppose X is infinite. We shall construct a total, computable, injective function $f : \mathbb{N} \rightarrow \mathbb{N}$ whose range is X . Precisely, we construct f such that, given input n , f returns the $(n+1)$ -th member of X in the sequence $0, 1, 2, 3, \dots$.

TM 1: φ_k : compute f

```

1 On input  $n$ :
2  $S \leftarrow \emptyset$ 
3 for  $i = 0, 1, 2, 3, \dots$  do
4   if  $\varphi_e(i) \downarrow$  then
5      $S \leftarrow S \cup \{\varphi_e(i)\}$ 
6     if  $|S| = n + 1$  then
7       output  $i$ 
```

□

Since X is infinite, $\varphi_k(n)$ will eventually halt and output the $(n+1)$ -th element of X for every $n \in \mathbb{N}$. Therefore, f is total.

Similarly, since φ_k only outputs an element of X with the condition that $\varphi_e(i)$ converges, every element outputted by f is a member of $\mathbf{dom}(\varphi_e) = W_e = X$. Therefore, $\mathbf{range}(f) = X$.

Finally, f is injective since it outputs a unique element for each $n \in \mathbb{N}$ since the set admits no duplicates.

\Leftarrow

Proof. For the two cases:

Suppose X is finite, then $X = \{x_1, x_2, \dots, x_n\}$ for some $n \in \mathbb{N}$. Define $g : \mathbb{N} \rightarrow \mathbb{N}$ by:

$$g(i) = \begin{cases} k & \text{if } i = x_k \text{ for some } k < n \text{ and } x_k \in X \\ \uparrow & \text{otherwise.} \end{cases}$$

Let e be the code of the turing machine that computes g . Then $\varphi_e(i) \downarrow$ *if and only if* $i \in X$. Therefore, $\mathbf{dom}(\varphi_e) = W_e = X$, so X is c.e.

Suppose X is infinite and it is the range of a total, computable, injective function $f : \mathbb{N} \rightarrow \mathbb{N}$.

We show that f is the domain of some function $g : \mathbb{N} \rightarrow \mathbb{N}$. Define g as follows:

$$g(n) = \begin{cases} i & \text{if } f(i) = n \\ \uparrow & \text{otherwise.} \end{cases}$$

TM 2: $g : \mathbb{N} \rightarrow \mathbb{N}$

```

1 On input  $n$ :
2 for  $i = 0, 1, 2, 3, \dots$  do
3   if  $f(i) \downarrow = n$  then
4     output  $i$ 
```

Let e be the code of the turing machine that computes g as outlined above. Then $\varphi_e(n) \downarrow$ *if and only if* $n = f(i)$ for some i , so $n \in \mathbf{range}(f) = X$. Therefore, $\mathbf{range}(f) = \mathbf{dom}(\varphi_e) = W_e = X$, and X is c.e. □

Problem 4.

Prove that K (the halting set) is **not** an index set.

1. $K = \{e \mid \varphi_e(e) \downarrow\}$.
2. An index set is a set X such that, for all e and k , if $\varphi_e = \varphi_k$ then $e \in X$ if and only if $k \in X$.

To show that K is not an index set, we shall find a code $e \in K$ and show that $k \notin K$ for some $\varphi_k = \varphi_e$.

Define a function f that converges only on its own code and diverges for all other $n \in \mathbb{N}$. That is, if e is the code of the machine that computes f , then

$$f(n) = \begin{cases} 1 & \text{if } n = e \\ \uparrow & \text{otherwise.} \end{cases}$$

We can do this because of the recursion theorem. Note that $e \in K$ since $\varphi_e(e) \downarrow$. By the *padding lemma*, for any e , there are infinitely many $k \neq e$ such that $\varphi_e = \varphi_k$. Pick one such k . What happens when we run $\varphi_k(k)$? Since $k \neq e$, $\varphi_e(k) \uparrow$ since $\varphi_e(k) \uparrow$. Therefore, $k \notin K$.

This means that K must not be an index set, since the condition

$$\varphi_e = \varphi_k \implies (e \in K \leftrightarrow k \in K)$$

does not hold.

Problem 5.

Show that if P is productive then P contains an infinite c.e. set.

Definition 5.1. A set P is productive if it has a productive function — a (partial) computable function ψ such that, whenever $W_e \subseteq P$, $\psi(e) \downarrow$ and $\psi(e) \in P \setminus W_e$. That is, a productive function is able to produce a witness to the fact that $P \neq W_e$ whenever $W_e \subseteq P$. Then it is immediate that productive sets are not c.e., so finding a c.e. set whose complement is productive will necessarily be a noncomputable c.e. set.

Let P be a productive function, with ψ as its productive function. We shall enumerate an infinite set $Y = \{y_0, y_1, y_2, \dots\} \subseteq P$ as follows:

1. Take e_0 to be the smallest index with $W_{e_0} = \emptyset \subseteq P$. Then $\psi(e_0) \downarrow = y_0$ for some $y_0 \in P \setminus \emptyset = P$.
2. Inductively, for $n \geq 1$, select e_n to be the smallest index such that $W_{e_n} = \{y_0, y_1, \dots, y_{n-1}\} \subseteq Y$.

Then $\psi(e_n) \downarrow = y_n$ for some $y_n \in P \setminus \{y_0, y_1, \dots, y_{n-1}\}$. Particularly, $y_n \neq y_i$ for any $i < n$.

To show that Y is c.e., we need to show that $Y = W_e$ for some e . Consider the following turing machine \mathcal{E}_Y that enumerates Y :

TM 3: $\varphi_e : Y \rightarrow \mathbb{N}$

```

1 On input  $n$ :
2 Initialize  $Y \leftarrow \emptyset$ 
3 for  $i = 0, 1, 2, \dots$  do
4   | Select  $e_i$  as above
5   | Compute  $y_i = \psi(e_i)$ 
6   | if  $y_i = n$  then
7   |   | output 1

```

On input n , the turing machine halts and outputs 1 if $n \in Y$. If $n \notin Y$, the turing machine will continue to loop, ad infinitum.

Therefore, **dom** $(\varphi_e) = Y$, so Y is c.e.