14. Rice's Theorem

One of many interesting applications of the ***recursion theorem*** is in the proof of the following result, which we will use to give many simple examples of sets that are ***non-computable***, and hence (by CT) ***undecidable***.

Recall the definition of the '$\sim$' relation on $\mathbb{N}$ (Sec. 13):

$$x \sim y \ =_{df} \ \varphi_x = \varphi_y.$$

Lemma 14.1. *The relation '$\sim$' is an **equivalence relation** on $\mathbb{N}$. Hence it partitions $\mathbb{N}$ into **equivalence classes**.*

A set $A \subseteq \mathbb{N}$ is called an ***index set*** iff A is closed under '$\sim$', i.e.,

$$\forall x, y \; (x \in A \wedge x \sim y \;\Rightarrow\; y \in A).$$

Now given sets $A \subseteq \mathbb{N}$ and $F \subseteq \text{COMP}$, let

$$\mathbb{F}(A) =_{df} \{\, \varphi_x \mid x \in A \,\} \subseteq \text{COMP},$$
$$\mathbb{I}(F) =_{df} \{\, x \in \mathbb{N} \mid \varphi_x \in F \,\} \subseteq \mathbb{N}.$$

So $\mathbb{I}(F)$ is the set of indices of functions in F. The two operations $\mathbb{F}$ and $\mathbb{I}$ are ***almost inverse*** to each other, in the following sense:

Lemma 14.2.

(a) *For any $F \subseteq \text{COMP}$, $\mathbb{F}(\mathbb{I}(F)) = F$.*

(b) *For any $A \subseteq \mathbb{N}$, $\mathbb{I}(\mathbb{F}(A)) = \{\, y \mid \exists x \in A\, (x \sim y) \,\}$,*
*i.e., the **closure** of A under '$\sim$'.*
Hence $\mathbb{I}(\mathbb{F}(A)) \supseteq A$, with equality iff A is an index set.

Corollary 14.3. *A subset of $\mathbb{N}$ is an index set iff it is the set of indices of some set of computable functions.*

EXAMPLES OF INDEX SETS:

1. $\mathbb{N}$
2. $\emptyset$
3. $[a] =_{df} \{\, b \mid b \sim a \,\}$, the '$\sim$'-***equivalence class*** of a, for any $a \in \mathbb{N}$
4. Any ***union*** of index sets.

Theorem 14.4 (Rice). *The only computable index sets are* $\mathbb{N}$ *and* $\emptyset$.

Proof: (J. Case): Suppose that

$$A \text{ is an index set,} \tag{1}$$
$$\emptyset \subset A \subset \mathbb{N}, \quad \text{and} \tag{2}$$
$$A \text{ is computable.} \tag{3}$$

We will now get a contradiction from (1), (2) and (3). By (2), choose

$$a \in A,\ b \notin A, \tag{4}$$

and define

$$f(x,z) \;\simeq\; \begin{cases} \varphi_b(x) & \text{if } z \in A \\ \varphi_a(x) & \text{if } z \notin A. \end{cases}$$

Then f is computable, since A is computable by (3).
By the ***recursion theorem***, there exists e such that

$$\varphi_e(x) \;\simeq\; f(x,e) \;\simeq\; \begin{cases} \varphi_b(x) & \text{if } e \in A \\ \varphi_a(x) & \text{if } e \notin A. \end{cases}$$

There are two possibilities:

$$\begin{array}{rl} & e \in A \;\Rightarrow\; \varphi_e = \varphi_b \;\Rightarrow\; e \sim b \;\overset{(1)}{\Rightarrow}\; b \in A, \\ \text{or} & e \notin A \;\Rightarrow\; \varphi_e = \varphi_a \;\Rightarrow\; e \sim a \;\overset{(1)}{\Rightarrow}\; a \notin A. \end{array}$$

Both possibilities lead to a contradiction to (4). $\square$

Corollary 14.5. *The following sets are **non-computable**:*

(a) $[a]$, *for any* $a \in \mathbb{N}$,

(b) $\{\, z \mid \varphi_z$ *total* $\}$,

(c) $\{\, z \mid \varphi_z$ *constant* $\}$,

(d) $\{\, z \mid \varphi_z$ *defined on at most finitely many arguments* $\}$,

(e) $\{\, z \mid \varphi_z$ *increasing* $\}$,

$\vdots$

NOTE:

By CT, Corollary 14.5(b) says that there is *no effective* method to decide, given any $\mathcal{G}$-program, whether it defines a *total function*. (This is related to the unsolvability of HP.)

In fact, by Section 8, Exercise 3 (p. 8-8), this problem is not even *semi-decidable*!

This shows that the notion of ***computable partial function*** (or ***partial*** algorithm) is ***more fundamental*** than the notion of ***computable total function*** (or ***total*** algorithm).

EXERCISES:

1. Prove Lemma 14.2 and Cor. 14.3.

2. (A uniform version of Exercise 3, p. 6-9): Show that there is a binary PR function f such that for all y, $\lambda n \cdot f(y,n)$ is 1-1, and for all y and n, $f(y,n) \sim y$.

3. Show that for every ***total computable*** f, there is a ***PR*** g such that for all x, $g(x) \sim f(x)$.

4. Is the relation '$\sim$' computable?

5. Let $f(x) =$ "the least y such that $y \sim x$". (Note that f is total.) Is f computable?