Math 260B — Mathematical Logic — Scribe Notes
UCSD — Spring Quarter 2012
Instructor: Sam Buss

Notes by: Tomoya Sato
April 11, 2012

1 The Halting Problem

Is there a Turing machine such that for any Turing machine M starting with a blank tape, it can compute whether or not M eventually halts? Is there a Turing machine such that for any Turing machine M with an input w, it can compute whether or not M eventually halts?

Definition 1. The Halting Problem is the set H

$$H := \{\ulcorner M \urcorner : M \text{ started with a blank tape and eventually halts.}\}$$

Theorem 1. *The Halting Problem H is undecidable.*

In order to show the theorem, we first show the following.

Definition 2. The M-w Halting Problem is the set H^* of pairs $(\ulcorner M \urcorner, w)$

$$H^* := \{(\ulcorner M \urcorner, w) : M(w) \text{ eventually halts.}\}$$

Theorem 2. *The M-w Halting Problem H^* is undecidable.*

Proof. (Proof by contradiction): Assume that N_1 is a Turing machine that decides H^*, i.e.,

$$N_1(\ulcorner M \urcorner, w) \text{ enters state } q_Y \quad \stackrel{\text{iff}}{\Longleftrightarrow} \quad M(w)\!\downarrow;$$

$$N_1(\ulcorner M \urcorner, w) \text{ enters state } q_N \quad \stackrel{\text{iff}}{\Longleftrightarrow} \quad M(w)\!\uparrow.$$

We modify N_1 to form another Turing machine N_2 such that

$$N_2(\ulcorner M \urcorner, w)\!\uparrow \quad \stackrel{\text{iff}}{\Longleftrightarrow} \quad M(w)\!\downarrow;$$

$$N_2(\ulcorner M \urcorner, w)\!\downarrow \quad \stackrel{\text{iff}}{\Longleftrightarrow} \quad M(w)\!\uparrow.$$

Finally, let N_3 be a Turing machine such that

$$N_3(\ulcorner M \urcorner) = N_2(\ulcorner M \urcorner, \ulcorner M \urcorner).$$

Then, we can derive a contradiction as follows:

$$N_3(\ulcorner N_3 \urcorner)\downarrow \overset{\text{iff}}{\iff} N_2(\ulcorner N_3 \urcorner, \ulcorner N_3 \urcorner))\downarrow \overset{\text{iff}}{\iff} N_3(\ulcorner N_3 \urcorner)\uparrow.$$

□

Definition 3. Let $Q, R \subseteq \Sigma^*$. A *many-one reduction from Q to R* is a (total) recursive function $f : \Sigma^* \to \Sigma^*$ such that for any $w \in \Sigma^*$,

$$w \in Q \iff f(w) \in R.$$

Theorem 3. *If R is decidable, then Q is decidable.*

Proof. Algorithm for deciding $w \in Q$ is the following: Input w. Then, compute $f(w)$ and check if $f(w) \in R$. If so, go to q_Y; otherwise, go to q_N. □

In order to show the Halting Problem H is undecidable, then, it suffices to show that there is a many-one reduction from the M-w Halting Problem H^* to the Halting Problem H. S defined as follows is the many-one reduction from H^* to H that we want:

$$S(\ulcorner M \urcorner, w) := \ulcorner M' \urcorner,$$

where M' is a Turing machine such that M' starts with the black input and

1. M' writes w on its input tape;

2. then it runs M.

(Notice that M' has ($|w| +$ the number of states in M)-many states.)

Many-one reductions are a special case of *Turing reductions*, which we will discuss in later class.

2 The Second Recursion Theorem

Let f be a partial recursive function with $(k+1)$ inputs, $X, w_1, \ldots, w_k \in \Sigma^*$.

Theorem 4. *There is a Turing machine such that for all input $\vec{w}$, $M(\vec{w}) = f(\ulcorner M \urcorner, \vec{w})$, i.e.,*

$$M(\vec{w})\downarrow \overset{\textit{iff}}{\iff} f(\ulcorner M \urcorner, \vec{w})\downarrow,$$

and if so, they give the same result.

Proof. Given f computed by some Turing machine M_0, form $g : \ulcorner N \urcorner \mapsto \ulcorner N' \urcorner$ such that $N'(\vec{w})$ computes $N(\ulcorner N \urcorner, \vec{w})$. (Notice, we still do not need a universal Turing machine here.) Suppose that g is computed by some M_1. We define h such that $h(\ulcorner N \urcorner, \vec{w}) = f(g(\ulcorner N \urcorner), \vec{w})$. We also suppose that h is computed by some M_2. Let M_3 be the Turing machine with Gödel number $\ulcorner M_3 \urcorner = g(\ulcorner M_2 \urcorner)$. Then,

$$\begin{aligned} M_3(\vec{w}) &= M_2(\ulcorner M_2 \urcorner, \vec{w}) \\ &= h(\ulcorner M_2 \urcorner, \vec{w}) \\ &= f(g(\ulcorner M_2 \urcorner), \vec{w}) \\ &= f(\ulcorner M_3 \urcorner, \vec{w}) \end{aligned}$$

□

Corollary 1. *There is a Turing machine M such that M starts with a blank tape and eventually outputs $\ulcorner M \urcorner$.*

Proof. Take f to be such that $f(X) = X$. □