CS 205

Lecture 35

12/4/21

Recall : $\langle M \rangle$: code of TM M | M_i : TM whose code is i

1. $L_d = \{ i \mid i \notin M_i \}$ is not r.e. Alt : $L_d = \{ \langle M \rangle \mid \langle M \rangle \notin L(M) \}$
2. $L_u = \{ \langle M \rangle \# w \mid M \text{ accepts } w \}$ is r.e. but not recursive.
3. $L_{halt} = \{ \langle M \rangle \# w \mid M \text{ halts on } w \}$ is r.e but not recursive.
4. $L_{ne} = \{ i \mid L(M_i) \text{ is non-empty} \}$ is r.e. but not recursive.
5. $L_e = \{ i \mid L(M_i) = \emptyset \}$ is not r.e.

Def : A property of r.e. languages is simply a set of r.e. languages. We express properties of r.e. languages as the sets of codes of TMs that accept those languages.

Ex 1. $\{ i \mid L(M_i) \text{ is infinite} \}$

2. $\{ i \mid L(M_i) \text{ does not contain a prime} \}$

Suppose M_1 and M_2 are two distinct TMs s.t. $L(M_1) = L(M_2)$. Then for a property P of r.e. languages either both or neither $\langle M_1 \rangle$ and $\langle M_2 \rangle$ belong to P.

Non-example : $\{ i \mid M_i \text{ has 15 states} \}$

— This is a property of TMs, not languages

Def : A property of r.e. languages is trivial if it is either empty or it is the set of all r.e. languages. Otherwise, it is called non-trivial.

Ex $\{ i \mid L(M_i) \text{ is r.e.} \}$ — trivial

$\{ i \mid L(M_i) \text{ is not r.e.} \}$ — trivial

$\{ i \mid L(M_i) = \emptyset \text{ or } L(M_i) \neq \emptyset \}$ - trivial

Observation : Trivial properties are decidable!

Rice's Thm : Every non-trivial property of r.e. languages is undecidable.

Examples of properties of TMs

$\{ i \mid M_i \text{ has 200 states} \}$ — decidable

$\{ i \mid M_i \text{ uses at most 35 tape cells on blank input} \}$ — decidable

$\{ i \mid M_i \text{ halts on blank input} \}$ — undecidable (exercise)

$\{ i \mid M_i \text{ on input 0011 at some point writes the symbol \$ on its tape} \}$ - undecidable

— Rice's thm. not applicable.

Proof of Rice's Thm

Let $L = \{ i \mid L(M_i) \text{ belongs to } P \}$ where

– at least one r.e. belongs to P, and one does not (i.e., P is non-trivial).

Plan : Reduce L_u to L. Since L_u is undecidable, so is L.

Assume, wlog, that the empty language $\emptyset$ does not belong to P. If it did, then we consider $\overline{L}$ and show that it is not recursive. Why?

Since P is non-trivial, at least one language is in P. Let $\boxed{i \in L}$, i.e. $L(M_i)$ belongs to P.

The reduction f from L_u to L :

Takes an instance $\langle M \rangle \# w$ for L_u and produces $f(M, w)$, the code of a TM s.t.

(i) If M accepts w then $M_{f(M,w)}$ accepts the same language as M_i

(ii) If M does not accept w then $M_{f(M,w)}$ accepts $\emptyset$. (Remember that by assump. $\emptyset$ does not belong to P)

Many-one reduction from L_1 to L_2
$f : \Sigma^* \to \Sigma^*$ s.t.
$w \in L_1 \iff f(w) \in L_2$

Thus $\boxed{\langle M \rangle \# w \in L_u \text{ iff } f(M, w) \in L.}$

Details of $M' = M_{f(M,w)}$

On input x

- Runs M on w
- If M does not accept or does not halt then do not accept x (or do not halt)
- If M does accept w then run M_i on x accept x iff M_i does.

Fact : If M accepts w then $L(M') = L(M_i)$

Otherwise $L(M') = \emptyset$.