Introduction to computability
Tutorial 9

Uncomputability

27 November 2018

Some undecidable languages

universal language: $UL = \{\langle M, w\rangle \mid M \text{ accepts } w\}$

$\overline{UL} = \{\langle M, w\rangle \mid M \text{ rejects or cycles on } w\}$

halting problem: $H = \{\langle M, w\rangle \mid M \text{ stops on } w\}$

empty-word halting problem: $\{M \mid M \text{ stops on } \varepsilon\}$

existential halting problem: $\{M \mid (\exists w)\ M \text{ stops on } w\}$

universal halting problem: $\{M \mid (\forall w)\ M \text{ stops on } w\}$

empty accepted language: $\{M \mid L(M) = \emptyset\}$

recursive accepted language: $\{M \mid L(M) \in R\}$

undecidable accepted language: $\{M \mid L(M) \notin R\}$

1. Let M_1 and M_2 be two Turing machines that accept respectively the languages L_1 and L_2.
Show that determining if $L_1 \subseteq L_2$ is undecidable.

1. Let M_1 and M_2 be two Turing machines that accept respectively the languages L_1 and L_2.
Show that determining if $L_1 \subseteq L_2$ is undecidable.

2. Show that $L \notin R$ and that $L \in RE$ where

$$L = \{M | M \text{ is a Turing machine and } \varepsilon \in L(M)\}.$$

1. Let M_1 and M_2 be two Turing machines that accept respectively the languages L_1 and L_2.
Show that determining if $L_1 \subseteq L_2$ is undecidable.

2. Show that $L \notin R$ and that $L \in RE$ where

$$L = \{M|M \text{ is a Turing machine and } \varepsilon \in L(M)\}.$$

3. Let L be a regular language and w a word.
The problem P determines whether or not there exists a word w' such that $ww' \in L$. Determine if $P \in R$ and if $P \in RE$. Justify.

4. Knowing that the problem of determining the universality of a context-free language (i.e. the problem of determining if for a context-free grammar G one has $L(G) = \Sigma^*$) is undecidable, prove that if G_1 and G_2 are two context-free grammars and G_R is a regular grammar, that the following problems are undecidable:

- $L(G_1) = L(G_2)$;
- $L(G_1) = L(G_R)$;
- $L(G_1) \subseteq L(G_2)$;
- $L(G_R) \subseteq L(G_1)$.

Some undecidable languages

universal language: $UL = \{\langle M, w\rangle \mid M \text{ accepts } w\}$

$\overline{UL} = \{\langle M, w\rangle \mid M \text{ rejects or cycles on } w\}$

halting problem: $H = \{\langle M, w\rangle \mid M \text{ stops on } w\}$

empty-word halting problem: $\{M \mid M \text{ stops on } \varepsilon\}$

existential halting problem: $\{M \mid (\exists w)\ M \text{ stops on } w\}$

universal halting problem: $\{M \mid (\forall w)\ M \text{ stops on } w\}$

empty accepted language: $\{M \mid L(M) = \emptyset\}$

recursive accepted language: $\{M \mid L(M) \in R\}$

undecidable accepted language: $\{M \mid L(M) \notin R\}$

5. An *unneeded* state of a Turing machine is a state that is never encountered, no matter what input word is considered.
Let M be a Turing machine and e a state of this machine. Show that the problem that consists of determining if the state e is unneeded by M is undecidable.

Bonus Exercise 10

Let M_1, M_2 and M_3 be Turing machines that accept respectively the languages L_1, L_2 and L_3.
Prove that the problem of determining if $(L_1 \cap L_2) \subseteq L_3$ is undecidable.