CSCI 335 Theory of Computing

Lecture 17 – Decidability III: The acceptance problem of TMs (A_{TM}) is undecidable

DATE : December 18 2022

Learning Outcomes

Learning Outcomes

By the end of this section you will be able to:

- Show that some languages are undecidable
- Show that the acceptance problem of TMs (A_{TM}) is undecidable

Outline

- The set of all infinite binary sequences are uncountable
- The set of all languages is uncountable
- The set of all Turing machines is countable
- Some languages are undecidable
- The acceptance problem of TMs (A_{TM}) is undecidable

The set of all infinite binary sequences are uncountable

The set of all infinite binary sequences are uncountable

(Exercise 4.7 page 211)
Let ***B*** be the set of all infinite sequences over {0,1}. Show that ***B*** is uncountable using a proof by diagonalization.

Proof: Suppose B is countable and a correspondence f : $N \longrightarrow B$ exists. We construct x in B that is not paired with anything in N. Let $x = x_1x_2 \ldots$. Let $x_i = 0$ if $f(i)_i = 1$, and $x_i = 1$ if $f(i)_i = 0$ where $f(i)_i$ is the ith bit of $f(i)$.
Therefore, we ensure that x is not $f(i)$ for any i because it differs from $f(i)$ in the ith symbol, and a contradiction occurs.

The set of all languages is uncountable

The set of all languages is uncountable

Let $\mathcal{L}$ = all languages

Corollary 1: $\mathcal{L}$ is uncountable

Proof: There's a 1-1 correspondence from $\mathcal{L}$ to $\mathbb{R}$ or ***B*** (the set of infinite binary sequences).

Let Σ^* = {s_1, s_2, s_3, . . .}. Each language A ϵ $\mathcal{L}$ has a unique sequence in ***B***. The i^{th} bit of that sequence is a 1 if s_i ϵ A and is a 0 otherwise. This sequence is called the ***characteristic sequence*** of A.

The function $f(A)$: $\mathcal{L}$ → ***B*** , is one-to-one and onto, and hence is a correspondence. Therefore, as ***B*** is uncountable, $\mathcal{L}$ is uncountable as well.

Σ^*	{ ε,	0,	1,	00,	01,	10,	11,	000,	...
$A \in \mathcal{L}$	{	0,		00,	01,				...
$f(A)$	.0	1	0	1	1	0	0	0	...

The set of all Turing machines is countable

The set of all Turing machines is countable

Theorem 2: The set of all Turing machines is countable

Proof: Let $\mathcal{M}$ = all Turing machines.

Since $\Sigma^* = \{\varepsilon, 0,1,00,01,10,11,000, \dots\}$ is countable, and the set $\{\langle M \rangle | \ M$ is a TM$\} \subseteq \Sigma^*$, then $\mathcal{M}$ is countable.

Some languages are undecidable

Some languages are undecidable

Corollary 2: Some language is not decidable.

Proof:

We proved that the set of languages $\mathcal{L}$ is uncountable, while the set of Turing machines is countable.

Thus, the set of languages $\mathcal{L}$ is larger than the set of Turing machines, i.e., there are more languages than TMs.

We conclude that some languages are not decidable (i.e., they have no decider Turing machines).

The acceptance problem of TMs (A_{TM}) is undecidable

The acceptance problem of TMs (A_{TM}) is undecidable

Recall $A_{\text{TM}} = \{\langle M, w\rangle \mid M$ is a TM and M accepts $w\}$

Theorem: A_{TM} is not decidable

Proof by contradiction: Assume some TM H decides A_{TM}.

So H on $\langle M, w\rangle = \begin{cases} \textit{Accept} & \text{if } M \text{ accepts } w \\ \textit{Reject} & \text{if not} \end{cases}$

Use H to construct TM D

$D =$ "On input $\langle M\rangle$

1. Simulate H on input $\langle M, \langle M\rangle\rangle$
2. *Accept* if H rejects. *Reject* if H accepts."

D accepts $\langle M\rangle$ iff M doesn't accept $\langle M\rangle$.

D accepts $\langle D\rangle$ iff D doesn't accept $\langle D\rangle$.

Contradiction.

The acceptance problem of TMs (A_{TM}) is undecidable

Why is this proof a diagonalization?

All TMs ⇩	All TM descriptions: $\langle M_1 \rangle$	$\langle M_2 \rangle$	$\langle M_3 \rangle$	$\langle M_4 \rangle$	$\ldots$	$\langle D \rangle$
M_1	acc	rej	acc	acc	$\ldots$	
M_2	rej	rej	rej	rej		
M_3	acc	acc	acc	acc	$\ldots$	
M_4	rej	rej	acc	acc		
$\vdots$						
D	rej	acc	rej	rej		?

Summary

Summary

- The set of all infinite binary sequences are uncountable
- The set of all Turing machines is countable
- Some languages are undecidable
- The acceptance problem of TMs (A_{TM}) is undecidable

Readings

- Undecidability: Section 4.2 (Sipser 2013)
- What is the halting problem? https://www.codingninjas.com/codestudio/library/halting-problem-in-the-theory-of-computation
- Halting problem https://en.wikipedia.org/wiki/Halting_problem