# A Quick Introduction to MATHEMATICAL LOGIC

SAEED SALEHI

Frontiers Summer School in Mathematics

Gödel's Incompleteness, 30 August 2021

# The Halting Problem (1)

Some Recursive Functions may Never Halt (may not have outputs on some inputs); e.g.,

$$D(x,y) = [\mu z.\, z+x=y] = \begin{cases} y-x & \text{if } x \leqslant y \\ \texttt{undefined} & \text{if } x > y \end{cases}$$

halts only when $x \leqslant y$.

**Notation**: $\begin{cases} f(x)\downarrow & f \text{ is defined at } x \\ f(x)\uparrow & f \text{ is } \textit{not} \text{ defined at } x \end{cases}$

Recursive Functions can be encoded by natural numbers:
Any description (proof) of a recursive function is a well-built sequence of $\langle Z, S, \pi_i^k, A, M, E, \chi_{\leqslant}, \wp, \circ, \boldsymbol{\mu}\rangle$ ($\circ$ stands for composition) and thus can be coded in $\mathbb{N}$.

Denote the (Gödel) code of the recursive function $f$ by $\ulcorner f \urcorner$.

# The Halting Problem (2)

### Theorem (Turing 1937)

*There is no recursive function* $\mathfrak{h}$ *such that for any* Recursive $f$,
$\mathfrak{h}(\ulcorner f \urcorner) = 1 \iff f(\ulcorner f \urcorner)\downarrow$ *and* $\mathfrak{h}(\ulcorner f \urcorner) = 0 \iff f(\ulcorner f \urcorner)\uparrow$.

### Proof.

Otherwise, $\mathfrak{g}(x) = \mu z.\, (z + \mathfrak{h}(x) = z)$ would be recursive too,
for which we have $\mathfrak{g}(\ulcorner f \urcorner)\downarrow \iff f(\ulcorner f \urcorner)\uparrow$ for every recursive $f$.
Putting $f = \mathfrak{g}$ we get the contradiction $\mathfrak{g}(\ulcorner \mathfrak{g} \urcorner)\downarrow \iff \mathfrak{g}(\ulcorner \mathfrak{g} \urcorner)\uparrow$ ! $\blacksquare$

### Corollary

*There is no algorithmic way for recognizing whether a given program is a virus (self-generating) or not.*

# An Undecidable, and a Non-Enumerable Set

## Corollary (The Halting Set is *Not* Decidable)

*The set of all (single-input) programs which halt on their own code is not decidable.*

$$\xrightarrow{\texttt{input}:\ \text{program } \mathcal{P}} \boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \begin{cases} \textbf{YES} & \text{if } \mathcal{P}(\ulcorner \mathcal{P} \urcorner)\downarrow \\ \textbf{NO} & \text{if } \mathcal{P}(\ulcorner \mathcal{P} \urcorner)\uparrow \end{cases}$$

## Theorem (The Halting Set *Is* Enumerable)

*An input-free algorithm enumerates the set* $\{\mathcal{P} \mid \mathcal{P}(\ulcorner \mathcal{P} \urcorner)\downarrow\}$.

## Proof.

Enumerate all the (single-input) programs $\mathcal{P}_0, \mathcal{P}_1, \cdots$.

Let $n := 1$; for $i = 0$ to $i = n$ run the $n$ stages of $\mathcal{P}_i(\ulcorner \mathcal{P}_i \urcorner)$; if it halts then PRINT "$i$"; let $n := n + 1$ and repeat. ∎

## Corollary (The Non-Halting Set is *Not* Enumerable)

*The set* $\{\mathcal{P} \mid \mathcal{P}(\ulcorner \mathcal{P} \urcorner)\uparrow\}$ *is not enumerable.*

# Decidable Structures

## Definition (Decision Problem for a Structure)

Fix a structure $\langle \mathbb{M}; \mathcal{L} \rangle$.

Input: a first-order $\mathcal{L}$-sentence $\varphi$. Output: $\begin{cases} \texttt{YES} & \text{if } \mathbb{M} \models \varphi \\ \texttt{NO} & \text{if } \mathbb{M} \not\models \varphi \end{cases}$

## Definition (Decidable Structure)

A structure is decidable if its decision problem is algorithmically solvable.

$$\xrightarrow{\texttt{input}:\ \varphi} \boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \begin{cases} \texttt{YES} & \text{if } \mathbb{M} \models \varphi \\ \texttt{NO} & \text{if } \mathbb{M} \not\models \varphi \end{cases}$$

# Enumerability in Structures

$$\boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \{\varphi_0, \varphi_1, \varphi_2, \cdots\} = \{\varphi \mid \mathbb{M} \vDash \varphi\}$$

**Theorem (Enumerable Structures are Decidable)**

*If $\mathbb{M}$ is an enumerable structure, then it is decidable.*

**Proof.**

If $\{\varphi \mid \mathbb{M} \vDash \varphi\}$ is enumerable, then so is its complement $\{\psi \mid \mathbb{M} \nvDash \psi\}$ because $\{\psi \mid \mathbb{M} \nvDash \psi\} = \{\psi \mid \mathbb{M} \vDash \neg\psi\}$. ■

$$\xrightarrow{\texttt{input}:\quad \varphi} \boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \begin{cases} \texttt{YES} & \text{if } \mathbb{M} \vDash \varphi \\ \texttt{NO} & \text{if } \mathbb{M} \nvDash \varphi \end{cases}$$

# Tarski's Theorems

## Theorem (Decidability of the Real (Ordered) Field)

*The structure* $\langle \mathbb{R}; 0, 1, -, \boldsymbol{\imath}', +, \times, \leqslant \rangle$ *is decidable.*

$$\xrightarrow{\texttt{input}:\ \text{ring-sentence } \varphi} \boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \begin{cases} \texttt{YES} & \text{if } \mathbb{R} \vDash \varphi \\ \texttt{NO} & \text{if } \mathbb{R} \nvDash \varphi \end{cases}$$

## Theorem (Decidability of the Complex Field)

*The structure* $\langle \mathbb{C}; 0, 1, -, \boldsymbol{\imath}', +, \times \rangle$ *is decidable.*

$$\xrightarrow{\texttt{input}:\ \text{ring-sentence } \varphi} \boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \begin{cases} \texttt{YES} & \text{if } \mathbb{C} \vDash \varphi \\ \texttt{NO} & \text{if } \mathbb{C} \nvDash \varphi \end{cases}$$

# Arithmetics of Presburger and Skolem

## Theorem (Presburger 1929)

*The structure* $\langle \mathbb{N}; 0, 1, +, \leqslant \rangle$ *is decidable.*

$$\xrightarrow{\texttt{input}:\ +\_\text{sentence } \varphi} \boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \begin{cases} \textbf{YES} & \text{if } \langle \mathbb{N}; + \rangle \vDash \varphi \\ \textbf{NO} & \text{if } \langle \mathbb{N}; + \rangle \nvDash \varphi \end{cases}$$

## Theorem (Skolem 1930)

*The structure* $\langle \mathbb{N}; 0, 1, \times \rangle$ *is decidable.*

$$\xrightarrow{\texttt{input}:\ \times\_\text{sentence } \varphi} \boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \begin{cases} \textbf{YES} & \text{if } \langle \mathbb{N}; \times \rangle \vDash \varphi \\ \textbf{NO} & \text{if } \langle \mathbb{N}; \times \rangle \nvDash \varphi \end{cases}$$

# Full Arithmetic $\langle \mathbb{N}; +, \times \rangle$

## Theorem (Gödel's Incompleteness 1931)

*The structure* $\langle \mathbb{N}; 0, 1, +, \times, \leqslant \rangle$ *is not decidable.*

$$\xrightarrow{\texttt{input}:\ (+,\times)\text{-sentence } \varphi} \boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \begin{cases} \texttt{YES} & \text{if } \langle \mathbb{N}; +, \times \rangle \vDash \varphi \\ \texttt{NO} & \text{if } \langle \mathbb{N}; +, \times \rangle \nvDash \varphi \end{cases}$$

## Corollary

*The structure* $\langle \mathbb{Z}; 0, 1, -, +, \times, \leqslant \rangle$ *is undecidable too.*

## Proof.

$\mathbb{N}$ is definable in it by the formula $0 \leqslant x$. ■

# THE END

**Corollary (J. Robinson 1949)**

*The structure* $\langle \mathbb{Q}; 0, 1, -, \imath', +, \times, \leqslant \rangle$ *is undecidable too.*

**Corollary**

*The structure* $\langle \mathbb{C}; 0, 1, -, \imath', e^x, +, \times \rangle$ *is undecidable too.*

---

**Problem (Open — Tarski)**

*Is the Real Exponential Field* $\langle \mathbb{R}; 0, 1, -, \imath', e^x, +, \times, \leqslant \rangle$ *decidable or not?*