

Logic, Computability and Incompleteness

Computability

Wolfgang Schwarz

24 October 2025

Midsemester Feedback LCI



PHILOSOPHY YEAR 4 FEEDBACK FORM

*Thank you so much! :)
Every response matters.
Help us enhance your
learning journey!*

Philosophy 4th Year Midterm Feedback



The Entscheidungsproblem

Hilbert's question (1928)

Is there an algorithm that decides validity for any first-order sentence?

Such an algorithm would take as input any sentence A and output

- **yes** if A is valid (true in all models);
- **no** if A is invalid (false in some model).

The Entscheidungsproblem

Brute-force proof search

Enumerate all finite strings; pick out those that are proofs of A.

- If A is valid, a proof appears eventually.
- If A is invalid, the search never halts.

This shows that validity is **semidecidable**.

We can mechanically enumerate all valid sentences.

Computable Functions

Computable Functions

Algorithms as functions

An algorithm is a procedure that computes an output from an input.

Every algorithm computes a function.

A function is **computable** if some algorithm computes it.

An algorithm may not halt on some inputs. Then it computes a merely partial function.

The Church–Turing Thesis

There are many ways to formalize the notion of algorithm.

- Turing machines
- Lambda-calculus
- (Partial) recursive functions
- ...

They are all equivalent.

Thesis: this matches our informal notion of mechanical computability.

Uncomputable Functions

Uncomputable Functions

The Halting Problem

Let $A_1, A_2, \dots$ be an effective enumeration of all algorithms.

Define the Halting function h by

- $h(n, x) = 1$ if A_n halts on input x ;
- $h(n, x) = 0$ otherwise.

No algorithm computes h .

Otherwise there would be an algorithm A_k that takes input n and

- halts if $h(n, n) = 0$,
- else loops forever.

$A_k(k)$ halts iff $h(k, k) = 0$ iff $A_k(k)$ does not halt.

Decidable and Semidecidable

Decidable and Semidecidable

A set/property/relation is **decidable** if its characteristic function is computable.

A set/property/relation is **semidecidable** (computably enumerable, r.e.) if there is an algorithm that lists exactly its elements.