Lecture #14: Oracle Reductions
Lecture Presentation

Consider the language $\text{LOOP}_{\text{TM}} \subseteq \Sigma_{\text{TM}}^{\star}$, including encodings of Turing machines M and input strings ω for M such that M ***loops*** on ω.

At this point in the course several similar languages have been considered:

- The language TM+I $\subseteq \Sigma_{\text{TM}}^{\star}$ of encodings of Turing machines M and input strings ω for M. This language is ***decidable*** and it follows from the definitions of these languages that $\text{LOOP}_{\text{TM}} \subseteq \Sigma_{\text{TM}}$.
- The language $\text{A}_{\text{TM}} \subseteq \Sigma_{\text{TM}}^{\star}$ of encodings of Turing machines M and input strings ω for M such that M accepts ω.

  This language is ***recognizable***: A multi-tape Turing machine with language A_{TM} (called a "universal Turing machine") was described in Lecture #12 — and it follows, by results about multi-tape Turing machines included in Lecture #10, that there must also exist a standard (single tape) Turing machine, $M_{\text{A}_{\text{TM}}}$, whose language is A_{TM}, as well.

  On the other hand it was proved in Lecture #13, that the language A_{TM} is ***undecidable***.

The goal of this lecture presentation will be to use an ***oracle reduction*** — along with the above information — to prove that the language LOOP_{TM} is also ***undecidable***.

Which Reduction Should We Use? Why?

An Algorithm That Uses a Subroutine

Adding Implementation-Level Details

How One Would Finish (If We Had Time and Wanted To Do Everything

Conclusion