Homework 7: Undecidability

CSE 30151 Spring 2022

Due: Thursday, April 14 at 11:59pm

Instructions

- Use this document to create a PDF file containing your solutions. Do this either by (1) printing this document, writing your solutions on it, and scanning[1] your work into a PDF; or (2) writing your solutions on this PDF digitally. Either way, please ensure that your work is clearly legible.
- If you need extra blank pages, feel free to insert them as needed. The amount of blank space provided beneath a problem does not necessarily indicate the expected length of your solution.
- You have the option to submit your solutions all at once or in parts; late penalties will only be applied to problems that are late. Individual problems cannot be submitted for grading in this way more than once.
- If you plan to submit some parts of your assignment late, before the deadline, upload a single PDF containing the problems you have completed so far. Do not include solutions to problems you want graded later for late credit. After the deadline, if you want to submit additional problems, add them to your original PDF and upload it again.[2]
- Submit your PDF file in Canvas under Assignments > Homework 7: Undecidability. You may re-submit your work any number of times before the due date.

[1] For tips on scanning your work using your mobile device, see `https://help.gradescope.com/article/0chl25eed3-student-scan-mobile-device`.

[2] For tips on concatenating your old and new PDFs together, see `https://help.gradescope.com/article/tp9kl4yx4q-student-troubleshooting-submissions`.

1. (a) Prove that it is undecidable whether a Turing machine M, on input w, ever attempts to move its head past the left end of the tape, using a reduction from another undecidable problem such as A_{TM}. (This is the same as Problem 5.14 [intl. 5.30] in the textbook.) **(4 points)**

(b) Prove that it is decidable whether a Turing machine M, on input w, ever attempts to move its head past the right end of the input string w. Provide a high-level description of a TM as your answer. (Hint: Re-read pages 221-223 of the textbook.) **(4 points)**

2. "The Power of 10" is a set of rules for writing mission-critical code developed at JPL.[3] Let us call a Turing machine that complies with these rules *10-compliant*. All you need to know about 10-compliance is:

   - A 10-compliant Turing machine always halts on every input.[4]
   - It is decidable whether a Turing machine is 10-compliant.

   In this problem, we'll show that any such set of rules will be incomplete in the sense that there is a language L_2 that is decidable, yet no TM that decides L_2 complies with the rules.

   Consider the language

   $$L_2 = \{\langle M \rangle \mid M \text{ is a 10-compliant TM that rejects } \langle M \rangle\}$$

   (a) Prove that L_2 is decidable. **(3 points)**

[3] `https://bit.ly/powof10`

[4] The rules also allow for a TM that never halts on any input; this is the desired behavior for, e.g., a daemon. But let's ignore this case.

(b) Prove that any TM that decides L_2 must not be 10-compliant. **(3 points)**

(c) Where in your solution to (a) is the violation of 10-compliance? (Full credit for any honest-effort answer; I just want you to think about it.) **(2 points)**

3. Let P be any property of Turing-recognizable languages that is nontrivial; that is, it isn't always true, and it isn't always false.

   Rice's Theorem (given in Problem 5.28 [intl. 5.16] of the textbook) says that it is undecidable, given a Turing machine M, whether $\mathcal{L}(M)$ has property P. Once you understand the statement of Rice's theorem, the following problems should be easy (don't overthink them).

   (a) Use Rice's Theorem to prove that it is undecidable whether a Turing machine M recognizes the language Σ^*. (Problem 5.30c, intl. 5.16) **(3 points)**

(b) Show that both conditions in Rice's Theorem are necessary (Problem 5.29 [intl. 5.17]) by

- showing that the two trivial properties (P is always false and P is always true) are decidable;
- giving an example of a non-trivial property of Turing machines – as opposed to the languages they recognize – that is decidable.

(5 points)

Changelog

- **Apr 14:** For the second bullet point of 3b, give an example of a *non-trivial* property of Turing machines.