Homework 7: Undecidability

Theory of Computing (CSE 30151), Spring 2024

Due: 2024-04-12 5pm

Instructions

- Create a PDF file (or files) containing your solutions. You can write your solutions by hand, but please scan them into a PDF.
- Please name your PDF file(s) as follows to ensure that the graders give you credit for all of your work:
  - If you're making a complete submission, name it `*netid*-hw7.pdf`, where *`netid`* is replaced with your NetID.
  - If you're submitting some problems now and want to submit other problems later, name it `*netid*-hw7-part123.pdf`, where `123` is replaced with the problem number(s) you are submitting at this time.
- Submit your PDF file(s) in Canvas.

Problems

In all of the following problems, any Turing machines that you write can be written as high-level descriptions.

1. **Bounds checking**
   (a) [Problem 5.30 (US ed. 5.14)] 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.
   (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. Your answer should be a high-level description of a TM.

2. ***The Power of 10*** is a set of rules for writing mission-critical code developed at JPL.[1] Let us call a Turing machine that complies with these rules *10-compliant*. All that you need to know about 10-compliance is:

[1] `http://bit.ly/powof10`

- A 10-compliant Turing machine always halts on every input.[2]
- 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.

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

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

3. **Rice's Theorem.** 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 (introduced in Problem 5.16 (US ed. 5.28), which you don't have to do) 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, then the following problems should be easy (don't overthink them):

   (a) [Problem 5.18 (US ed. 5.30c)] Use Rice's Theorem to prove that it is undecidable whether a Turing machine M recognizes the language Σ^*.

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

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

[2] 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.