CS3800: Theory of Computation — Summer II ’22 — Drew van der Poel

Homework 5
Due Saturday, August 13 at 11:59pm via Gradescope

Name:
Collaborators:

- Make sure to put your name on the first page. If you are using the LaTeX template we provided, then you can make sure it appears by filling in the `yourname` command.
- This assignment is due Saturday, August 13 at 11:59pm via Gradescope. No late assignments will be accepted. Make sure to submit something before the deadline.
- Solutions must be typeset. If you need to draw any diagrams, you may draw them by hand as long as they are embedded in the PDF. I recommend using the source file for this assignment to get started.
- I encourage you to work with your classmates on the homework problems. *If you do collaborate, you must write all solutions by yourself, in your own words.* Do not submit anything you cannot explain. Please list all your collaborators in your solution for each problem by filling in the `yourcollaborators` command.
- Finding solutions to homework problems on the web, or by asking students not enrolled in the class is strictly forbidden.

Problem 1. *Multi-Automata Decidability* (8 points)

For each of the following languages, state whether or not they are decidable. If a language is decidable, give a proof (description of an algorithm) showing why. If not, give an argument as to how this would contradict one of the undecidability results we saw in class.

(a) **[4 pts.]** $L_1 = \{\langle P, T, w\rangle |$ P is a PDA, T is a Turing Machine, w is an input string and at least one of P or T accepts w$\}$

Solution:

(b) **[4 pts.]** $L_2 = \{\langle N, G\rangle |$ N is a NFA, G is a CFG, and $L(N) = L(G) = \emptyset\}$

Solution:

Problem 2. *Proving Undecidability with Reducibility (*8 points*)*

Consider the following language,
$L_5 = \{\langle M \rangle | \ M$ is a Turing machine and there is a string of length 5 in $L(M)\}$

Prove that L_5 is undecidable with a reducibility argument, using A_{TM}, the Turing machine-acceptance problem.

Solution:

Problem 3. *Diagonalization (*4 points*)*

Recall the emptiness problem for Turing machines, E_{TM}. We showed in class that E_{TM} is undecidable via being reducible from A_{TM}. Now you're asked to construct an alternative proof of E_{TM} being undecidable, using a *diagonalization argument*.

Solution:

Problem 4. *We've Seen This Language Before....* (6 points)

Recall from earlier, $L_5 = \{\langle M \rangle |\ M$ is a Turing machine and there is a string of length 5 in $L(M)\}$. Prove that $\overline{L_5}$ is unrecognizable. You should not use a reduction in your proof.

Solution:

Problem 5. *Thought Experiment (*4 points*)*

In class we saw how to reduce A_{TM} to $HALT_{TM}$ to show $HALT_{TM}$ was undecidable. Suppose the roles were reversed, you know $HALT_{TM}$ is undecidable and want to use proof by contradiction to show A_{TM} is also undecidable.

Note that this is a thought experiment and the application doesn't make sense, as we used the undecidability of A_{TM} to prove that $HALT_{TM}$ was undecidable. So for our purposes (if it makes you happier), you can imagine $HALT_{TM}$ has somehow (magically) been shown to be undecidable but A_{TM} has not.

(a) **[1/2 pt.]** We are proving by contradiction. What is your initial assumption? What does this imply in terms of the existence of some Turing machine(s)? What is the input to that Turing machine?

Solution:

(b) **[1/2 pt.]** What are the possible outcomes of your above Turing machine on some input and what does each outcome mean?

Solution:

(c) **[2 pts.]** At least one of the outcomes above isn't helpful in concluding anything about $HALT_{TM}$. Identify any such outcome and explain why it doesn't allow us to draw conclusions about $HALT_{TM}$?

Solution:

(d) **[1 pt.]** Part (c) tells us this proof wouldn't work. But for the sake of exercise, for each remaining outcome from part (b), explain why this does allow us to draw conclusions about $HALT_{TM}$.

Solution: