
This homework is due 24 hours later than usual, on *Thursday 2/22* at 8pm. This is because some of the questions (labeled below) rely on the lecture on 2/19.

This homework has 8 questions, for a total of 100 points and 5 extra-credit points.

Unless otherwise stated, each question requires *clear, logically correct, and sufficient* justification to convince the reader.

For bonus/extra-credit questions, we will provide very limited guidance in office hours and on Piazza, and we do not guarantee anything about the difficulty of these questions.

We strongly encourage you to typeset your solutions in L^AT_EX.

If you collaborated with someone, you must state their name(s). You must *write your own solution* for all problems and *may not use any other student's write-up*.

(0 pts) 0. **Before you start; before you submit.**

If applicable, state the name(s) and username(s) of your collaborator(s).

Solution:

(10 pts) 1. **Self assessment.**

Carefully read and understand the posted solutions to the previous homework; you may also find the video “walkthroughs” on Canvas helpful. Identify one part for which your own solution has the most room for improvement (e.g., has unsound reasoning, doesn’t show what was required, could be significantly clearer or better organized, etc.). Copy or screenshot this solution, then in a few sentences, explain what was deficient and how it could be fixed.

(Alternatively, if you think one of your solutions is significantly *better* than the posted one, copy it here and explain why you think it is better.)

If you didn’t turn in the previous homework, then (1) state that you didn’t turn it in, and (2) pick a problem that you think is particularly challenging from the previous homework and explain the answer in your own words. You may reference the answer key, but your answer is to be in your own words.

Solution:

2. **A powerful diagonalization.**

- (7 pts) (a) The *power set* of a set A , denoted $\mathcal{P}(A)$, is defined as the *set of all subsets* of A . For example, $\mathcal{P}(\{0, 1\}) = \{\emptyset, \{0\}, \{1\}, \{0, 1\}\}$.

Use diagonalization to prove that for any *countably infinite* set A , the power set $\mathcal{P}(A)$ is *uncountably infinite*.

Hint: View each subset as an infinite binary sequence representing whether each element of A is included in the subset or not.

Solution:

- (6 pts) (b) Prove that any infinite language $L \subseteq \Sigma^*$ has an undecidable subset $L' \subseteq L$.
Hint: Part (a) will be useful.

Solution:

3. Turing machines.

- (7 pts) (a) Go to <https://turingmachine.io/> and step through the example code that adds 1 to an input integer (represented as a binary string). Modify this machine to add 2 instead of 1 (scroll down on the page for instructions on how to write the code). Give a picture or screenshot of your new machine as your solution. You do not need to provide justification.

Solution:



- (8 pts) (b) Let L be the language of all binary strings that represent non-negative integers that are divisible by 4:

$$L = \{\langle s \rangle : s \bmod 4 = 0\} \cup \{\varepsilon\}.$$

Note that leading zeroes in the representation are acceptable, i.e., 000101010 and 101010 both represent 42. Also, we define ε to represent 0.

Draw a state-machine diagram for a Turing machine that decides L . This drawing can be done however you like (hand-drawn, using <https://turingmachine.io/>, drawn in latex, etc.), as long as it is readable. No justification is necessary.

Hint: How is the last bit of the binary representation of an integer related to whether the integer is divisible by 2? Can you extend this observation to divisibility by 4?

Solution:

4. Decidability.

- (7 pts) (a) Prove or disprove: any finite language is decidable.

Solution:

- (5 pts) (b) Let $L = (L_1 \setminus L_2) \cap L_3$, where L_1, L_2 , and L_3 are *decidable* languages. State whether L is decidable for **all**, **some** (but not all), or **no** such L_1, L_2, L_3 , and prove it.
Note: $A \setminus B := \{x \in A : x \notin B\}$ denotes the *set difference* between A and B .

Solution:

- (7 pts) (c) Let $L = L_1 \cap L_2$, where L_1 and L_2 are both *undecidable*. State whether L decidable for **all**, **some** (but not all), or **no** such L_1, L_2 , and prove it.

Solution:

5. **Haltchecker.** (Warning: This problem is related to material from the lecture on 2/19.)

- (8 pts) (a) Recall from lecture the definition $L_{\text{HALT}} := \{(\langle M \rangle, x) : M \text{ halts on input } x\}$. In class you saw a Turing reduction from L_{ACC} to L_{HALT} , proving that L_{HALT} is undecidable.
[This linked video](#) describes a different argument that L_{HALT} is undecidable, without using a reduction.
Formalize the argument in the video as a rigorous proof.

Solution:

- (7 pts) (b) Consider the following language:

$$L_{\text{HALT} < 2^{|x|}} = \{(\langle M \rangle, x) : M(x) \text{ halts within } 2^{|x|} \text{ steps}\}.$$

Prove or disprove: $L_{\text{HALT} < 2^{|x|}}$ is decidable.

Solution:

6. **Turing reductions.** (Warning: This problem relies on material from the lecture on 2/19.)

Before embarking on this problem, read [Handout 2: Turing Reductions](#).

- (7 pts) (a) Prove or disprove: if $L \leq_T L'$ for some language L' , then L is decidable.

Solution:

- (7 pts) (b) Prove or disprove: $L \leq_T L'$ for any decidable language L and any language L' .

Solution:

- (7 pts) (c) Define

$$L_{\text{AND-LOOP}} = \{(\langle M \rangle, x, y) : M \text{ is a TM that loops on input } x \text{ and loops on input } y\}.$$

Prove that $L_{\text{HALT}} \leq_T L_{\text{AND-LOOP}}$. Note that since L_{HALT} is undecidable, this means that $L_{\text{AND-LOOP}}$ is undecidable.

Solution:

- (7 pts) (d) Let $L_{\text{SELF-LOOP}} = \{\langle M \rangle : M \text{ is a TM that loops on input } \langle M \rangle\}$. Prove that $L_{\text{SELF-LOOP}}$ is undecidable, via a Turing reduction from a problem that we proved undecidable in class.

Solution:

- (5 EC pts) 7. **Optional extra credit: An interesting language.**

Recall that a language L is *recognizable* if there is a Turing machine that accepts every string in L , and either rejects *or loops* on every string not in L .

Prove that there exists a language L where neither L nor its complement \bar{L} is recognizable.

Solution: