Martin Ziegler
Ivan Adrian Koswara

Issued on: March 18, 2019
Solutions due: March 25, 2019

CS422

Spring 2019, Homework #2

Problem 3 (5 × 5 points)

Fill each box with one of ✓ for "yes", ✗ for "no", or ? for "unknown". Give a brief reasoning for each item. The first two rows have been filled as examples. (You are recommended to print this page. If your need more space, you may put your reasons on a separate sheet.)

Decidable	Semi-decidable	Statement (assume inputs are given in some suitable encoding)
✗	✓	The halting problem, i.e. "given algorithm $\mathcal{A}$ and input x, does $\mathcal{A}$ halt when run on input x?" **Reason:** *Proven in class; this is semi-decidable but not decidable.*
?	✓	range(f) for a computable, injective, total function $f: \mathbb{N} \to \mathbb{N}$ **Reason:** *This is the definition of enumerable languages. Enumerable languages are always semi-decidable but may or may not be decidable.*
		The set of natural numbers that can be represented as the sum of two prime numbers **Reason:**
		The problem "given algorithm $\mathcal{A}$ and inputs x, n, is it true that, when $\mathcal{A}$ is run on input x, it enters infinite loop *or* it halts within n steps?" **Reason:**
		The problem "given algorithm $\mathcal{A}$ and inputs x, n, is it true that, when $\mathcal{A}$ is run on input x, it enters infinite loop *or* it does *not* halt within n steps?" **Reason:**
		range(f) for a computable, strictly increasing, total function $f: \mathbb{N} \to \mathbb{N}$ **Reason:**
		L where $\overline{L}$ is semi-decidable **Reason:**

(Problems 4 and 5 on next page)

Problem 4 (4 + 4 + 4 points)

Consider the function $\langle\cdot\rangle: \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ defined as $\langle x, y\rangle = 2^x(2y+1) - 1$. (Note that this is different from the one introduced in the lecture.)

a. Prove that $\langle\cdot\rangle$ is injective.
b. Prove that $\langle\cdot\rangle$ is surjective.
c. Sketch an algorithm[1] that computes the inverse of $\langle\cdot\rangle$. You don't need to prove its correctness.

Problem 5 (6 + 6 + 6 points)

Recall that we have discussed the decision problems H, N, T in class. Consider the following decision problem, called E for "equivalence":

> "Given two algorithms $\mathcal{A}, \mathcal{B}$, is it true that for each input x, $\mathcal{A}$ halts when run on x if and only if $\mathcal{B}$ halts when run on x (not necessarily in the same number of steps)?"

Prove the following items by constructing an explicit reduction and proving that the reduction is correct.

a. $\overline{N} \preccurlyeq T$.
b. $T \preccurlyeq E$.
c. $E \preccurlyeq T$.

[1] The algorithm can be in any language of your choice. It can even be in pseudocode, simply describing the general steps without the exact implementation details.