Prep Work 12 - Computability
CS 234
Daniel Lee

1. Computability

1. In your own words, what is the Church-Turing thesis?

Any computational model is based on or could be performed by a Turing machine.

2. Can your computer accept any languages that a Turing machine cannot?

No. All modern computers are no more or less powerful than a Turing machine, and conversely, modern computers can perform any computation that a Turing machine can. Therefore, any language that a Turing machine cannot accept also cannot be accepted by computers either.

3. In your own words, what is the Universal Turing machine?

A Turing machine that is capable of running a simulation of other Turing machines based on the given input string and the encoding of a Turing machine.

4. In your own words, give an intuitive idea of how the Universal Turing machine works.

Based on storing the input string on one tape, the current state on another, the head position on a third, and the Turing machine encoding on a fourth, we can use these tapes to identify what character is at the current head position and what state we are currently at and then search the Turing machine encoding tape for the correct move to execute next during each step of the simulation.

5. In your own words, what is the definition of a recursive language?

A language that has a Turing machine that accepts it and halts on all inputs.

6. Are recursive languages closed under complement? Why or why not?

Yes, because any language L is recursive if and only if the complement of L is recursive.

7. In your own words, what is the definition of a recursively enumerable language?

A language that has a Turing machine that accepts it.

8. In your own words, give an intuitive idea of how the proof of Theorem 14.2 works.

The proof of Theorem 14.2 works by first showing that if the language L is recursive, then L and the complement of L are both recursively enumerable with using the properties of Theorem 14.1 and the fact that L has a Turing machine and thus L is also recursively enumerable by definition.

Then we show that by assuming L and the complement of L are both recursively enumerable and then running the simulations for a Turing machine M and the complement of M in parallel based on the existence of the Universal Turing Machine.

9. In your own words, give an intuitive idea of how the proof of Theorem 14.3 works.

The proof of Theorem 14.3 works based on proof by contradiction, assuming that the complement of the language D is accepted by a Turing machine with encoding i. Then the proof observes that the Turing machine M_i cannot be a Turing machine for the complement of D in either case of whether M_i accepts i or rejects i.