Prep Work 13 - Reductions

CS 234

Daniel Lee

1. Computability

1. Look at Theorem 14.5. It takes the form of a proof by contradiction. What is assumed for the sake contradiction, and how is that property used by the proof? (Hint: it gives access to a Turing machine satisfying some particular properties.)

The proof assumes that the language A is recursive. This property is used in the proof by deducing the property that there exists a Turing machine M for the language A that halts on all inputs.

2. A reduction between two machines A and B shows that you can reduce the behaviour of A to that of B. In other words, it is a program for A that uses B as a subroutine. What is the reduction that occurs in Theorem 14.5?

The reduction is demonstrated by using the Turing machine M for A to the Turing machine M' for D. This tells us that if the language D were recursive, the language A would also be recursive.

3. Remember, limitations on Turing machines are also limitations on other forms of computing like Python programs. What does Theorem 14.5 mean in terms of Python programs?

There are no Python programs that can determine the output of an arbitrary Python program on arbitrary input for that arbitrary Python program.

4. Look at Theorem 14.6. It takes the form of a proof by contradiction. What is assumed for the sake contradiction, and how is that property used by the proof?

The proof assumes that the language *SAME* is recursive. This property is used in the proof by deducing the property that there exists a Turing machine M for *SAME* that halts on all inputs.

5. What is the reduction that occurs in Theorem 14.6?

The reduction is demonstrated by using the Turing machine M for *SAME* to the Turing machine M' for D that halts on all inputs. This tells us that if the language D were recursive, the language *SAME* would also be recursive.

6. What does Theorem 14.6 mean in terms of Python programs?

There are no Python programs that can determine if two arbitrary Python programs behave identically on all inputs.

7. Look at Theorem 14.7. It takes the form of a proof by contradiction. What is assumed for the sake contradiction, and how is that property used by the proof?

The proof assumes that the language H is recursive. This property is used in the proof by deducing the property that there exists a Turing machine M for the language H that halts on all inputs.

8. What is the reduction that occurs in Theorem 14.7?

The reduction is demonstrated by using the Turing machine M for H to the Turing machine M' for D that halts on all inputs. This tells us that if the language D were recursive, the language H would also be recursive.

9. What does Theorem 14.7 mean in terms of Python programs?

There are no Python programs that can determine if an arbitrary Python program has infinite loop bugs.