EECS 343: Theoretical Computer Science, Homework Exercise 3
due Monday, February 10, 2020 before class

Problem 1: The International Obfuscated C Code Contest (a real thing) gives a desired output for a program, and programmers compete to see who can write the hardest to read code that achieves the output. This year's challenge (not true), is to write code that produces "EECS343 is fun". Let's prove that detemining if every entry actually achieves that is really hard!

Let $L_{343} = \{\langle M \rangle \mid M \text{ is a Turing machine and } \mathcal{L}(M) = \{EECS343, is, fun\}\}$. That is, L_{343} is the set of machines that accept the strings "EECS343", "is", and "fun", and no other strings. Prove that L_{343} is undecidable.

Problem 2: The problem of obsolete code is well known in software engineering. When code is maintained and updated, some lines of code may become obsolete in that they will never be executed. It would be nice to be able come up with a technique that will always identify and remove these lines of code. On a Turing machine, we will represent obsolete code as a *useless state*. You will show that it is impossible to always recognize useless states.

A *useless state* in a Turing machine is one that is never entered on any input string. Consider the problem of identifying whether a Turing machine has any useless states. Formulate this problem as a language and show that it is undecidable.

Problem 3: Give an example of an undecidable language B where $B \leq_m \overline{B}$, and prove the reduction.