Exercise Set 7: Reduction Junction, what's your function?

The first thing you should do in exercise7.tex is set up your name as the author of the submission by replacing the line, `\submitter{TODO: your name}`, with your name and UVA email id, e.g., `\submitter{Grace Hopper (gmh1a)}`.

Before submitting , also remember to:

- List your collaborators and resources, replacing the TODO in `\collaborators{TODO: replace ...}` with your collaborators and resources. (Remember to update this before submitting if you work with more people.)
- Replace the second line in exercise7.tex, `\usepackage{uvatoc}` with `\usepackage[response1]{uvatoc}`, `\usepackage[response2]{uvatoc}`, etc. for the appropriate problem submission.

Collaborators and Resources: TODO: replace this with your collaborators and resources (if you did not have any, replace this with *None*)

Exercise 7-1: The next Gr8 programming language

Professor Brunelle has written his own programming language which he calls N8. Brunelle is particularly excited for this programming language because it has the fantastic ability to let you know when you've written a program that might have an infinite loop! Specifically, when you compile a program written in N8, and there's at least one input that causes the program to enter an infinite loop, the compiler lets you know (and it doesn't warn you when there is no such input).

Show that the N8 programming language is less powerful than a Turing Machine. That is, show that there must be at least one Turing machine that cannot be converted into an equivalent N8 program.

Exercise 7-2: Re-Enters

Show that the function $REENTERS$ defined below is uncomputable.

> **Input:** A string w that describes a Turing Machine.
>
> **Output:** **1** if the machine described by w would re-enter its start state when executed on the input ε. Otherwise, **0**.

That is, a machine which computes $REENTERS$ outputs **1** when the input describes a Turing Machine which, when run with the input ε, enters the start state as a result of some transition.

Exercise 7-3: Rice's Theorem

For each subproblem, indicate whether Rice's Theorem applies. If it applies, explain why, and answer if the problem is computable or uncomputable. If it does not apply, just indicate why it doesn't apply (it is not necessary to determine whether or not it is computable if Rice's theorem does not apply).

(a) Given the description of a Turing Machine, does that machine always return 0.

(b) Given the description of a Turing Machine, does that machine always return 1 when it receives no input.

(c) Given the description of a Turing Machine, does that machine ever use more than 10,000 cells on its tape.

(d) Given the description of a Turing Machine, is the language of that machine recognizable.

(e) Given the description of a Turing Machine, does the Turing machine have exactly 50 states.

(f) For a fixed way of describing Turing machines, does the string $110100111100010100010010011100100101$ describe a Turing machine which accepts 101?

