Exercise Set 4: Automata Tune

The first thing you should do in exercise4.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 exercise4.tex, `\usepackage{uvatoc}` with `\usepackage[response2]{uvatoc}`, `\usepackage[response3]{uvatoc}`, `\usepackage[response4]{uvatoc}`, `\usepackage[response5]{uvatoc}` 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 4-2: Indescribability

We mentioned in class that a set S is closed under some operation provided that applying the operation to elements of S is guaranteed to always result in an element from S. For example, the set of odd numbers is closed under multiplication because the product of two odd numbers is odd. The set of odd numbers is not closed under addition, as the sum of two odd numbers may be even (in fact it always is, but to not be closed there only needs to be one counterexample).

This problem is intended to be an exercise in the skill of abstract reasoning, as well as to help you to become familiar with closure. For this reason, we've intentionally asked you to reason about a set for which by definition we cannot give you any example members.

There are only a finite number of languages spoken by humans. Any text written or spoken in any of those languages is going to be finite in length. This means that there is only a countable number of things that can be described in a way humans can understand. There are an uncountable number of real numbers. These together imply that there are some real numbers that cannot be written down in a way that any human can understand it. We will call the set of all such numbers the indescribable numbers.

Determine whether or not the set of indescribable numbers is closed under the following operations (i.e. determine whether or not that operation applied to indescribable numbers is guaranteed to always result in a number that is indescribable). Accompany your answer with a proof:

(a) **negation**: the negation of x is $-x$

(b) **addition**

(c) **inversion**: the inverse of x is $\frac{1}{x}$

(d) **square root**

Exercise 4-3: Regexes

Give a regex for each of the following languages:

1. $\{x \in \{0,1\}^* | x$ when representing a natural number is divisible by $8\}$
2. $\{x \in \{0,1\}^* | x$ does not contain the substring $011\}$
3. $\{x \in \{0,1\}^* |$ every odd position of x is a $1\}$

Exercise 4-4: Sugary Regexes

In class, our definition of regular expressions could only use the empty string (i.e. "" or ε), string literals (e.g. a or b), concatenation, alternation, or Kleene star. In practice, regular expressions may have other components/operations as well. For each of the following operations allowed in regular expressions in practice, show that it is merely syntactic sugar. That is, show that you can convert any regular expression that uses those components into one using only those from class.

(a) **Question Mark**: If R is a regular expression, then the regular expression $R?$ will match either the empty string or a string which matches R. For example, $ab?$ matches the strings a and ab.

(b) **Plus**: If R is a regular expression, then the regular expression $R+$ will match any string composed of 1 or more substrings which match R. For example, $(a|b)+$ matches the strings a, b, ab, ba, etc., but not ε.

(c) **Count**: If R is a regular expression, and n is a natural number, then $R\{n\}$ will match any string composed of exactly n substrings which match R. For example, $(0|1)\{9\}$ will match any binary string of length 9.

(d) **Count Range**: If R is a regular expression, and m, n are natural numbers with $m > n$, then $R\{m - n\}$ will match any string composed of between n and m (inclusive) substrings which match R. For example, $b\{2 - 4\}$ will match bb, bbb, and $bbbb$.

(e) **Character Class**: A character class gives a set of characters, and matches any 1 character from that set. If a, b, c are characters from the alphabet, then the character class $[abc]$ will match a, b, and c.

Exercise 4-5: Closure with a language operation

We showed in class that computability by finite state automata is closed under complement. This means that for any language which I could compute using a finite state automaton, I could also compute its complement using a finite state automaton. The way we did this was to show that I could take any given finite state automaton, and manipulate it in some way to produce a finite state automaton which returned 1 for exactly all of the strings for which the original machine returned 0. Show that computability by finite state automata is also closed under the `OneShorter` operation defined below. To do this, show how I could take any finite state automaton (say it computes language $L \subseteq \Sigma^*$), and manipulate it to produce a finite state automaton which computes the language $\texttt{OneShorter}(L)$.

$$\texttt{OneShorter}(L) = \{x \in \Sigma^* | \exists a \in \Sigma . xa \in L\}$$

In English, the language $\texttt{OneShorter}(L)$ is the set of all strings x for which there is at least one character a such that adding a to the end of the string x would make it a string in L. In other words, it is the set of all strings from L with their last characters removed.

