Assignment 01: Regular Languages

Computer Science 311 Spring 2016

Due April 19th, 2016

Problem 1 When writing decimal integers that may be long, we often separate the digits into groups of three (e.g. "15,679") to make them easier to parse visually.

Draw a state diagram for a finite automaton that takes a string and determines whether it represents a valid decimal integer with group separators and *without* leading zeroes. Then, provide a formal specification of this machine.

Problem 2 Show that any finite language is a regular language.

Problem 3 Show that the regular languages are closed under intersection.

Problem 4 In C, the atof() function is used to parse an array of char elements into a float. If the input string doesn't represent a valid floating-point number, though, the function simply returns 0.0. This means that if atof() returns 0.0, we don't have a way to determine whether the user entered a representation of zero or an invalid string.

Write a regular expression that we can use to validate user input when atof() returns 0.0. Remember that -0 = 0.

Problem 5 If A and B are regular languages, define $A \diamond B = \{xy \mid x \in A \text{ and } y \in B \text{ and } |x| = |y|\}.$

Use the Pumping Lemma to show that regular languages are *not* closed under this operation.

Problem 6 (Half Credit)

