

**CMSC 330 homework on regular expressions and DFA**

**DUE: Wednesday, October 4. Turn in either in class or in office hours**

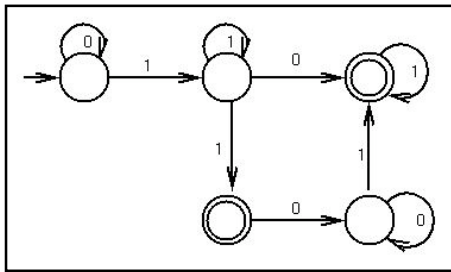
1. Give regular expressions for the following languages. You should only use formal regular expressions as defined on the slides.

- (a) Binary strings ending in 01
- (b) Decimal integers divisible by 5. Initial 0's are not allowed (e.g., 015 is invalid, but 15 is a valid string; 0 is also a valid string).
- (c) Ruby identifiers. (You'll need to find out what valid Ruby identifiers are on your own.)
- (d) Binary strings consisting of either an odd number of 0s or an odd number of 1s.
- (e) Binary strings containing the sequence 101 embedded within it.
- (f) Binary strings that do not contain the sequence 101 embedded within it.

2. Give DFA for:

- (a)-(f) in (2) above
- (g) Decimal integers divisible by 9. (Hint: Look up casting out 9s) The DFA for this problem might be easier written down in formal notation (with a table for the transition function) rather than drawing a picture.

3. Give the regular expression that accepts the same set as the following NFA:



4. Given the set  $L = \{a^{3m}b^{2n} \mid m \geq 0; n \geq 0\}$

- (a) Describe  $L$  as a regular expression.
- (b) Give a DFA that recognizes this set.
- (c) Does the regular expression  $a^*b^*$  generate  $L$  since every string in  $L$  can also be generated by  $a^*b^*$ ? Explain.

5. Let  $S$  be a set recognized by a DFA. Prove that  $S^R$  ( $S$ -reversed, i.e., every string in  $S$  written in reverse order) is recognized by a DFA.

6. Let  $R$  be a set of even length strings generated by a DFA. Let  $R^{1/2}$  be the “first halves” of all strings in  $R$ . That is, if  $w$  is in  $R$  and  $w$  is of length  $2n$ , then the first  $n$  symbols of  $w$  is recognized by  $R^{1/2}$ . Show that  $R^{1/2}$  is recognized by a DFA. (Hint 1: Solve problem 5 first. Hint 2: Run a DFA backward and forward at the same time.)

Note: This is a hard problem. Few will probably get this one correct, but everyone should attempt it. Hint 2 should make the problem a bit easier to solve.