Homework 2: DFAs and NFAs

CSE 30151 Spring 2017

Clarifications added 2017/01/31
Due 2017/02/02 at 11:55pm

Instructions

- Create a PDF file (or files) contaning your solutions.
- Please name your PDF file(s) as follows:
  - If you're making a complete submission, name your PDF file `netid-hw2.pdf`, where `netid` is replaced with your NetID.
  - If you're submitting some problems now and want to submit other problems later, name your PDF file `netid-hw2-123.pdf`, where `123` is replaced with the problems you are submitting at this time.
- Submit your PDF file in Sakai. Don't forget to click the Submit (or Resubmit) button!

Problems (10 points each)

1. **Designing finite automata.** Write a finite automaton for natural numbers (written in base-10; **you choose whether `0` is allowed and whether leading zeros are allowed; empty string is not allowed**) that are:

   (a) divisible by 2. Please write both a formal description **and** a state diagram.

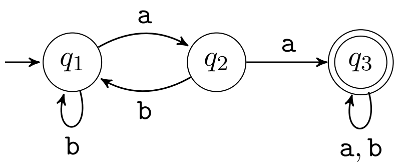
   (b) divisible by 3. Please write both a formal description **and** a state diagram.

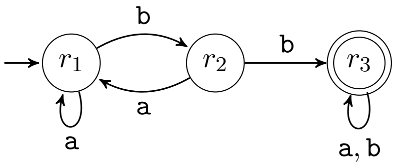
   (c) divisible by k for any given $k > 0$. Your answer should be a formal description $M = (Q, \Sigma, \delta, s, F)$ where Q, δ, s and F are defined in terms of k. Hint: appending a digit d to a number x is equivalent to doing $x \leftarrow 10x + d$.

2. **Boolean operations.** Define $L_1 \uparrow L_2 = (L_1 \cap L_2)^C$ (that is, the NAND operation on languages).

   (a) Use a product construction, as in the proof of Theorem 1.25, to prove that regular languages are closed under the $\uparrow$ (NAND) operation.

   (b) Apply your construction to the following two DFAs:





(c) A *Boolean function* is a function from k Boolean values to a Boolean value. For example,

x	y	$f(x,y)$
0	0	1
0	1	0
1	0	1
1	1	1

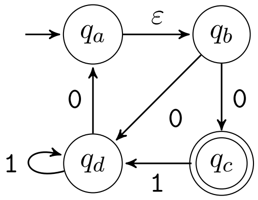
Recall that any Boolean function can be expressed in terms of $\uparrow$. For example, the above function can be written as $f(x, y) = (x \uparrow x) \uparrow y$. Using this fact, show that regular languages are closed under any Boolean function. That is, if $L_1, \ldots, L_k$ are regular languages, and f is a function from k Boolean values to a Boolean value, then the language $f(L_1, \ldots, L_k)$ is regular:

$$w \in f(L_1, \ldots, L_k) \text{ if and only if } f(w \in L_1, \ldots, w \in L_k).$$

(Here "$x \in X$" is being used as a Boolean value, which is a little nonstandard for mathematical writing, but I hope the meaning is clear.)

3. **Nondeterminism**

(a) Convert the following NFA into an equivalent DFA.



(b) Write an NFA N that recognizes the following language:

$$L = \{uv \mid u, v \in \{\texttt{a}, \texttt{b}\}^*, u \text{ contains an even number of } \texttt{a}\text{'s, and}$$
$$v \text{ contains an even number of } \texttt{b}\text{'s}\}$$

(c) Show, for any $n \geq 0$, what the accepting path for $\texttt{bab}^n$ through N would be. **It's okay to show the path for $n = 0, 1, 2, 3, \ldots$ until the pattern is clear.**

(d) Convert N to a DFA M, and again show, for any $n \geq 0$, what the accepting path for $\texttt{bab}^n$ through M would be. **It's okay to show the path for $n = 0, 1, \ldots$ until the pattern is clear.**