CS 374 Lab 3: DFA Proofs: Correctness and Lower Bounds

Date: January 26, 2016.

Recall that a fooling set for a language $L \subseteq \Sigma^*$ is a set of strings $F \subseteq \Sigma^*$ such that for every $u, v \in F$ with $u \neq v$, there is $w \in \Sigma^*$ such that either $uw \in L$ and $vw \notin L$ or $uw \notin L$ and $vw \in L$. Further if a language L has a fooling set of size k then any DFA recognizing L must have at least k states.

Problem 1. [Category: Proof] Consider the language L_1 that consists of all strings that have 1 in every odd position. The DFA for L (that you designed in Lab 2) is shown in Figure 1

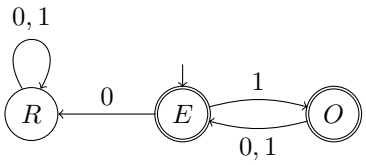

Figure 1: DFA for problem 1

1. Prove that the DFA in Figure 1 correctly recognizes L.
2. Prove that any DFA recognizing L has at least 3 states.

Problem 2. [Category: Design] Design a DFA for the language $A_k = \{bbc^n \,|\, n \bmod k = 0\}$ over the alphabet $\{b, c\}$.