6.045J/18.400J: Automata, Computability and Complexity Prof. Nancy Lynch

Recitation 2: DFAs and NFAs

February 12, 2003 *Susan Hohenberger*

Problem 1: Define the following words, phrases and symbols.

1. Finite state machine, finite automaton
2. Sipser page 49. Determinism vs Nondeterminism
3. Sipser page 54. DFA vs NFA
4. Regular Language
5. Sipser page 53. $(Q, \Sigma, \delta, q_0, F)$
6. ϕ
7. ϵ
8. Epsilon Transition
9. Sipser page 58. Regular Operations
10. Sipser page 36,40. A machines can accept many strings, but only a single language. To avoid confusion, we will usually say a machine accepts a string and recognizes a language.

Problem 2: Are the following statements true or false?

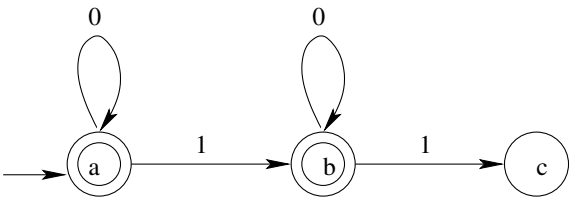
1. It is possible for a finite automaton to recognize an infinite language.
2. Every deterministic finite automaton is also a nondeterministic finite automaton.
3. For every nondeterministic finite automaton, there is an equivalent nondeterministic finite automaton that has no epsilon transitions.
4. For every nondeterministic finite automaton, there is an equivalent nondeterministic finite automaton that has a single accept state.
5. If you swap the accept states and the reject states on ANY finite automaton, the new machine will recognize the complement of the original language.
6. The class of languages recognized by non-deterministic finite automata is closed under complementation.
7. Sipser problem 1.24. For any string $w = w_1 w_2 \dots w_n$, the **reverse** of w, written w^R, is the string w in reverse order, $w_n \dots w_2 w_1$. For any language A, let $A^R = \{w^R | w \in A\}$. If A is regular, then so is A^R.

Problem 3: Create finite automata for each of the following languages over the alphabet $\Sigma = \{0, 1\}$. Give a characterization for each state.

1. The language of strings that contain both a '0' and a '1'.
2. The language of strings that contain '0110' and end in '1'.

3. The language of strings that contain a '0' and don't end in '10'.
4. The language of strings that do not contain an odd number of 1s.
5. The language of binary numbers that are divisible by 5.
6. The language of part 2 concatenated with that of part 3.
7. The language of part 1 star.

Problem 4: Let's prove that the following automaton recognizes exactly the language $L = \{w \in \{0,1\}^* | w$ contains less than 2 ones$\}$. To do this, we will need to prove that our FA (1) accepts all strings in L and (2) does not accept any string not in L.



1. Characterize each state.
2. Forward direction (accepts all strings in L). Proof by Induction?
3. Reverse direction (does not accept any string outside of L). Proof by Contradiction?