1MD3 Tutorial 8 – Finite State Machine

CAO Shiqi

March 8, 2006

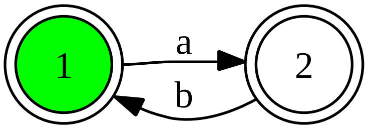
1 Quick Review

Finite State Machine is an abstract computer with limited number of memory. It contains the following components,

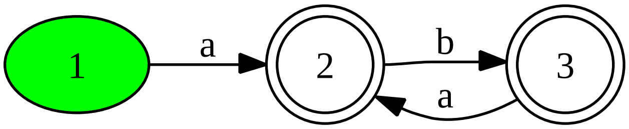
- a set of alphabet
- a set of states
- a set of accepting states, this set must be a subset of all states
- a transfer function
- a start state, which must be a member of the set of states

2 Some Examples and Exercises

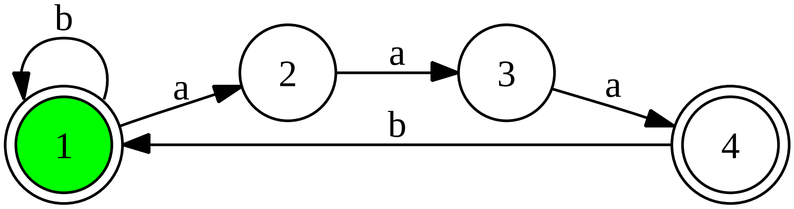
Example 1: Let the alphabet set be $\{a, b\}$, design a FMS which accepts any string such that a occurs on all odd number of position in the string. The position starts from 1. Such that $\{ababab\ldots\}$



Example 2: Let M be the language accepted by the FSM in the example 1. Let M' be $M - \{\lambda\}$. Design a FSM to accept M'.



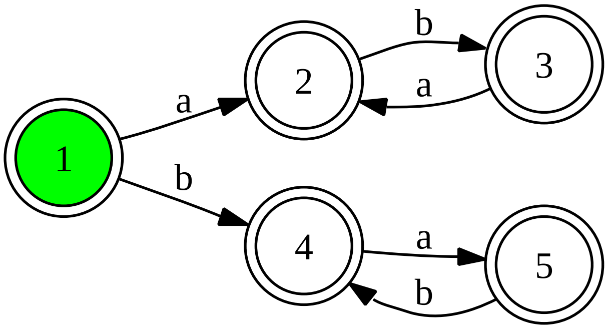
Example 3: Design a FSM to accept any string such that in the string, if a occurs, it is always in a group of three. For instance, $\{baaabbbbbbaaabaaabb\}$



Exercise 4: Design a FSM to accept any string such that in the string a always occurs in a group. The group size is between 1 to 3. For instance, $\{babbbbbbaabaaabb\}$

TRY IT ON YOUR OWN

Example 5: Design a FSM to accept any string such that in the string a and b are alternating. $\{\lambda\}$ is also accepted. For example, $\{babab\}$,$\{ababab\}$.



Exercise 6: Redo the Example 3 with one state less.

Exercise 7: Redo the Example 5, but only use three states.