



## Theory of Computing

Ali Fele Paranj  
alifele@student.ubc.ca

*September 26, 2025*







## 0. Contents

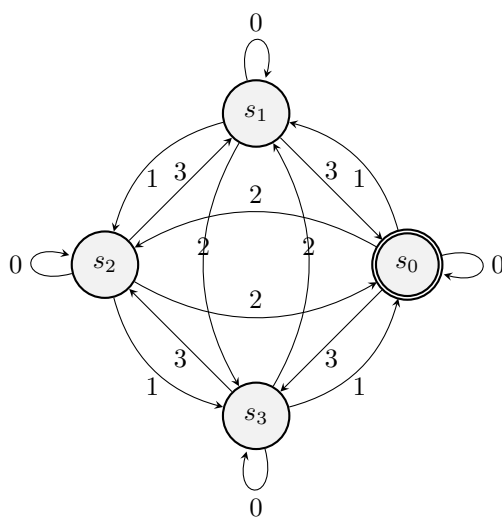
1 Regular Languages

5



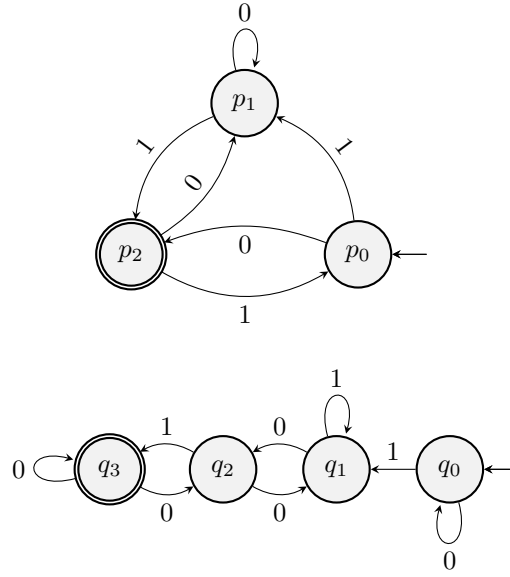
# 1. Regular Languages

**Observation 1.1** The transition rules of a finite automaton can formally shown as  $\Sigma \times S \rightarrow S$ . When  $\Sigma$  is a group, then some transition rules are actually group actions (satisfies the conditions for being a group action). When  $\Sigma = S = G$  for some group, then interesting stuff can happen. For instance, let  $\Sigma = S = \mathbb{Z}_4$  cyclic group of order 4. Then the resulting finite automata can accept the strings that their sum is mod 4. Then the transition rules will be as below. This finite state machine can accept the strings that their numerical sum of the values of the string is 0 modulo 4.



## Cartesian product of finite automaton

■ **Example 1.1** Let  $M_1$  and  $M_2$  be finite automata, and let  $L_1$  and  $L_2$  be the languages that they recognize respectively. Then  $M_1 \oplus M_2$  is the finite automaton that recognizes  $L_1 \cup L_2$ . For instance, consider the following FSM's:



The state space of  $M_1 \oplus M_2$  is  $S_1 \times S_2$  and the transition functions is  $\delta = \delta_1 \otimes \delta_2$ , given by

$$\delta((p, q), s) = \delta_1 \otimes \delta_2((p, q), s) = (\delta_1(p, s), \delta_2(q, s)).$$

■