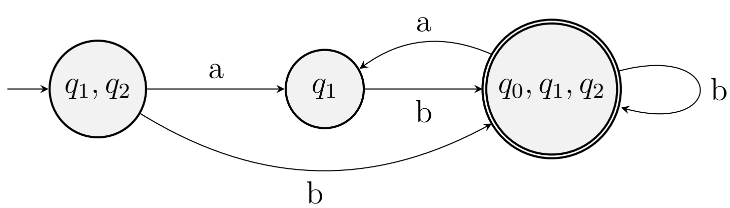
Homework Assignment #1

Steven Au
NLP 201
Flanigan

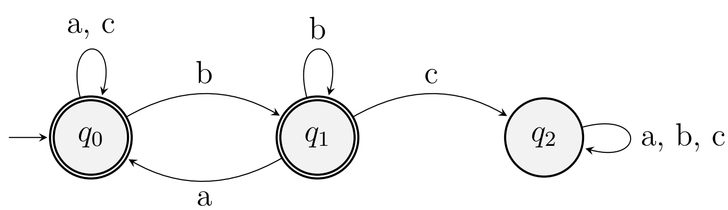
October 24, 2023

Problem 1

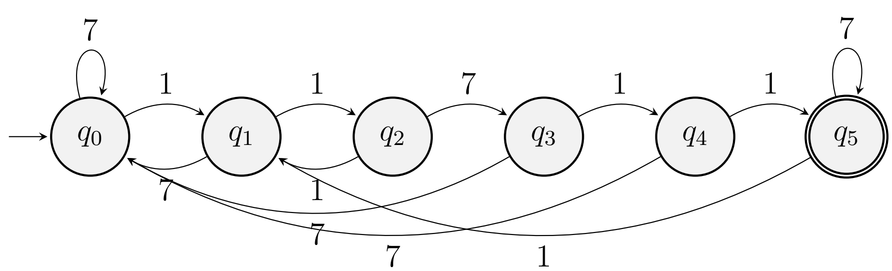


Problem 2

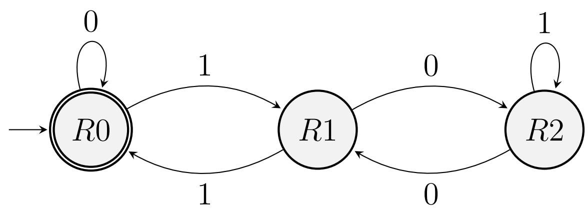
a) The set of strings over the alphabet {a, b, c} in which the substring *bc* never occurs.



b) The set of strings over the alphabet {1, 7} that end in 11711.

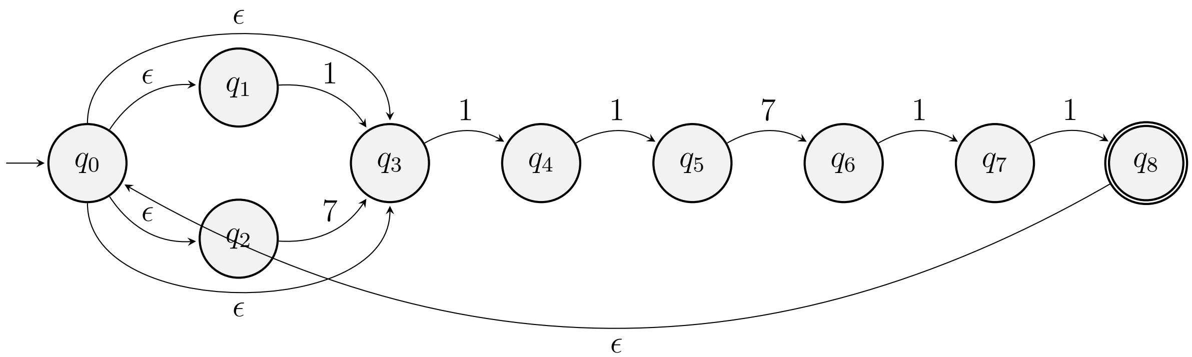


c) The set of strings over the alphabet $\{0, 1\}$ which are divisible by three when interpreted as a binary number (ignoring leading zeroes).



Problem 3

The set of strings over the alphabet $\{1, 7\}$ ending in 11711.



The NFA does not need to be deterministic and we can remove a lot of the edges.

Problem 4

a) The set of strings over $\{a, b, c\}$ in which the substring bc never occurs.

$$c^*(b|ac^*)^*$$

b) The language described in Problem 2.3: The set of strings over $\{0, 1\}$ which are divisible by three when interpreted as a binary number (ignoring leading zeroes).

$$0^*(1|(01^*0)^*1)^*$$

Problem 5

We may define generalized regular expressions (GREs) as follows:

1. $\emptyset$ is a GRE denoting the empty language.
2. ε is a GRE denoting the language $\{\varepsilon\}$.
3. For each $\sigma \in \Sigma$, σ is a GRE denoting the language $\{\sigma\}$.

4. If α and β are GREs, denoting the languages A and B, respectively, then
   - $(\alpha \,|\, \beta)$ is a GRE denoting $A \cup B$.
   - $(\alpha\beta)$ is a GRE denoting $A.B$.
   - α^* is a GRE denoting A^*.

   new $(\alpha \wedge \beta)$ is a GRE denoting $A \cap B$.

   new $\neg\alpha$ is a GRE denoting $\bar{A}$.

1. **Base Cases**: $\emptyset$ and ε are regular by definition. Any finite language is regular, and both of these represent finite languages.

2. Same case as 1.

3. **Finite Automaton (FA) Construction**: Create a DFA that only accepts strings of length 1 containing the symbol σ and rejects all other strings in a trap state. Therefore a DFA can be constructed for any single symbol from the alphabet proving a language is regular.

4. **Closure Properties**: Regular languages are closed under certain operations like union, intersection, complement, concatenation, and Kleene star.
   - **Union** $(\alpha \,|\, \beta)$: If A and B are regular languages, then their union $A \cup B$ is also regular.
   - **Concatenation** $(\alpha\beta)$: If A and B are regular, then their concatenation $A \cdot B$ is also regular.
   - **Kleene Star** (α^*): If A is regular, then its Kleene star A^* is regular. The Kleene star represents zero or more concatenations of the language with itself.
   - **Intersection** $(\alpha \wedge \beta)$: If A and B are regular, then their intersection $A \cap B$ is also regular. Create a deterministic finite automaton (DFA) from the DFAs that accept A and B with product construction. The resultant DFA accepts an input if and only if both original DFAs accept the input.
   - **Complement** $(\neg\alpha)$: If A is regular, then its complement is regular. Swapping the accepting and non-accepting states of the DFA that recognizes A, results in a DFA will accept exactly the strings that the original DFA rejected.

Problem 6

1. **False.** Counterexample: $L_1 = \{a^n b^n \,|\, n \geq 0\}$ is not regular. $L_2 = \Sigma^*$ is regular and contains L_1.

2. **False.** Counterexample: For $L_1 = \Sigma^*$ (all strings) and $L = \emptyset$, L_2 can be any language, including non-regular ones.

3. **False.** Counterexample: With $L_1 = \emptyset$ and $L = \Sigma^*$, L_2 can be anything, even non-regular.

4. **False.** Counterexample: For $L_i = \Sigma^* - \{a^i\}$, each L_i is regular. The infinite intersection excludes all strings a^n, which is not regular.

5. **True.** Explanation: If $\beta|\alpha\gamma$ is in $L(\gamma)$, then $L(\alpha^*\beta)$ must be in $L(\gamma)$.