Regular Expressions

Sipser: Section 1.3 pages 63 - 69

\[(0 \cup 1)0^*\]

\[(0 \cup 1)^*\]

\[\Sigma^*1\]

\[R^*\]

Long Ago in a Place Not Far Away

Old Home Week
**Regular Expressions**

**Definition.** Say that \( R \) is a *regular expression* if \( R \) is

1. \( a \) for some \( a \) in the alphabet \( \Sigma \),
2. \( \varepsilon \),
3. \( \emptyset \),
4. \( (R_1 \cup R_2) \), where \( R_1 \) and \( R_2 \) are regular expressions,
5. \( (R_1 \circ R_2) \), where \( R_1 \) and \( R_2 \) are regular expressions,
6. \( (R_1)^* \), where \( R_1 \) is a regular expression.

**Working with Regular Expressions**

\[
0^*10^* = \{ w | \}
\]

\[
= \{ w | w \text{ is a string of odd length} \}
\]

\[(0 \cup \varepsilon)(1 \cup \varepsilon) = \]

\[(01)^*\emptyset = \]

\[(\ast \cup - \cup \varepsilon)(DD^* \cup DD^* \cup D^* \cup D^* \cup DD^*) = \]

where \( D = \{0,1,2,3,4,5,6,7,8,9\} \)

**Identities**

Let \( R \) be a regular expression.

- \( R \cup \emptyset = \)
- \( R \circ \varepsilon = \)
- \( R \cup \varepsilon = \)
- \( R \circ \emptyset = \)

**Regular Expressions and NFAs**

**Theorem.** A language is regular if and only if some regular expression describes it.

**Proof.** (\( \Leftarrow \))

1. If \( a \in \Sigma \), then \( a \) is regular.
2. \( \varepsilon \) is regular.
3. \( \emptyset \) is regular.
4. If \( R_1 \) and \( R_2 \) are regular, then \( (R_1 \cup R_2) \) is regular.
5. If \( R_1 \) and \( R_2 \) are regular, then \( (R_1 \circ R_2) \) is regular.
6. If \( R_1 \) is regular, then \( (R_1)^* \) is regular.
Build an NFA that recognizes the regular expression: \((ab \cup a)^*\)

Build an NFA that recognizes the regular expression: 
\(a(a \cup b)^a\)