

Regular Expressions

Definition of Regular Expressions

- A regular expression on some alphabet Σ is defined inductively as:
 - \emptyset , ϵ , and $a \in \Sigma$ are regular expressions.
 - If P and Q are regular expressions, then $P \cup Q$, $P \circ Q$, and P^* are regular expressions.
- Examples

Regular Expressions and Finite Automata

- Theorem:

A language is regular iff some regular expression describes it.

– Regular expression \implies regular language

– Regular language \implies regular expression

Applications of Regular Expressions

- Describing patterns
 - awk, grep, perl
 - vi, emacs
- Regular expressions in compilers

Converting DFAs into Regular Expressions

1. Convert DFAs into GNFA's

GNFA: generalized nondeterministic finite automaton

- transition arrows: any regular expressions as labels.
- A special form of GNFA (for convenience)
 - the start state: has arrows to all other states, but no in-coming arrow.
 - only a single accept state: has arrows from all other states, but no out-going arrows.
 - other states: at most one arrow between any two states, and to each state itself.

2. Convert GNFA's into regular expressions

Convert GNFA's into Regular Expressions

- Remove the states in GNFA one at a time.
- CONVERT (G), G is a k -state GNFA
 - If $k = 2$, done.
 - If $k > 2$, select any state (except the start and accept states), remove it and modify the GNFA correspondingly.
- Example

The Big Picture

- Equivalent models of computation
Regular languages \iff DFAs \iff NFAs \iff regular expressions

DFAs with Output

- Moore Machines $(Q, \Sigma, \Delta, \delta, q_0, \lambda)$
 - Δ : output alphabet
 - $\lambda: Q \rightarrow \Delta$,
giving the output associated with each state.
 - DFA, but no final states.
- Example

DFAs with Output

- Mealy machines $M = (Q, \Sigma, \Delta, \delta, q_0, \lambda)$
 - Δ : output alphabet
 - $\lambda: Q \times \Sigma \rightarrow \Delta$,
giving the output associated with the transition from state q on input a .
- Example
- Equivalence of Mealy and Moore machines

Next Time

- Pumping Lemma
- Nonregular Languages