##### EQUIVALENCES WITH REGULAR LANGUAGES

**Regular expressions, REs, deterministic finite automata, DFAs, and non-deterministic finite automata, NFAs, are all equivalent in their descriptive power even though superficially appear to be rather different. Recall that a regular language is one that is recognized by some deterministic finite automaton.**

**THEOREM 1.28**

***A language is regular iff some regular expression describes it.***

**This theorem has two directions. We state and prove each direction as a separate lemma.**

**(Now, to complete our cycle we will establish the equivalence between regular expression R and a NFA which recognizes L(R)).**

**LEMMA 1.29:**

**If a language is recognized by a regular expression, then it is regular.**

**PROOF IDEA:**

**Say that we have a regular expression R describing some language A. We show how to convert R into a NFA recognizing A. By Corollary 1.20, if an NFA recognizes A then A is regular.**

**PROOF:**

**Let’s convert R into an NFA N.**

**We consider the six cases in the formal definition of regular expressions.**

**1. R = a for some a in . Then L(R) = {a}, and the following NFA recognizes L(R).**

a

**Formally,**

**N = ({q1, q2}, , q1, {q2}), where we describe  by saying that**

**(q1, a) = {q2},  (r, b) = Ø for r ≠ q1 or b ≠ a.**

**Note:**

**We could have presented an equivalent DFA here but a NFA is all we need for now, and it is easier to describe.**

**2. R = **. Then L(R) = {**}, and the following NFA recognizes L(R).**

**Formally,**

**N = ({q1}, , q1, {q1}), where  (r, b) = Ø for any r and any b.**

**3. R = Ø. Then L(R) = Ø, and the following NFA recognizes L(R).**

**Formally,**

**N = ({q}, , q, Ø), where  (r, b) = Ø for any r and any b.**

**4. R = R1 U R2**

**5. R = R1° R2**

**6. R = R1\***

**The last three cases use the constructions given for regular languages.**

#### Example

**Convert the regular expression (ab ∪ a)\* to an NFA.**

**a**

a

**b**

b

**ab**

a

b

******

**ab ∪ a**

a

b

******

a

******

******

**(ab ∪ a )\***

******

a

b

******

a

******

******

******

******

#### Example

**Convert the regular expression *(a ∪ b)\*aba* to an NFA.**

**(a ∪ b)\***

******

b

a

******

******

******

******

**(a ∪ b)\*aba**

******

b

a

******

******

******

******

a

b

******

******

a

******

******

******