CS 311: Computational Structures

James Hook

October 7, 2014

3 Closure Properties

3.1 Recall

- Construction of a DFA from an NFA
- Proof that the DFA and NFA accepted the same language

3.2 Plan

- NFA and DFA equivalence (conclusion)
- Discuss Problem Set 1 (particularly closure under reverse)
- Closure under Union (Cartesian product construction)
- Closure under Concatenation
- Closure under the star operation
- Regular Expressions

3.3 NFA and DFA equivalence

Last lecture we showed that given an NFA N we can construct an equivalent DFA M.

How do we convert a DFA to an NFA?

Showing that we can freely convert between these computational models shows that they define the same family of languages. We characterize that with the theorem:

Theorem 3.1 *A language is regular iff it is recognized by an NFA.*

3.4 Problem Set 1 Discussion

3.5 Closure Properties

In this section we see how to combine Regular languages to form new regular languages.

We say a set is closed under an operation if whenever we apply the operation to elements of the set we get another element of the set. For example, the positive numbers are closed under addition, but not under subtraction. The integers are closed under negation, addition, subtraction, and multiplication.

In the case of closure properties for computational models, we will show that families of languages are closed under language level operations. For example, in Problem Set 1 we demonstrated that the family or regular languages is closed under the reverse operation.

3.5.1 Complement

Warm up exercise. The regular languages are closed under complement.

First, what do we mean? Given $A \subseteq \Sigma^*$ the complement of A is the set of strings not in A. This is written $\bar{A}$

$$\bar{A} = \{x \in \Sigma^* | x \notin A\}$$

How do we show closure under complement? We give a construction.

Given a regular language A we know there is a DFA $M = (Q, \Sigma, \delta, q_0, F)$ such that $A = \mathcal{L}(M)$. The DFA $M' = (Q, \Sigma, \delta, q_0, Q - F)$ recognizes $\bar{A}$.

3.5.2 Union

DFA product construction.

NFA construction.

3.5.3 Concatenation

3.5.4 Star

3.5.5 Intersection

3.6 Regular Expressions

Definition. (Example of inductive)

Examples.

Thm: Every language described by a REGEXP is regular

What about the other way?