**Air Force Institute of Technology**

**Graduate School of Engineering and Management**

**Department of Electrical and Computer Engineering**

**CSCE 532 Automata and Formal Languages**

**Winter 2019**

# Day 6

# nonregular Languages

# Context-Free Grammars

§1.3 Regular Expressions and Nonregular Languages (cont.)

### Practice (Sipser Exercise 1.46a)

Prove that is not regular. You may use the pumping lemma and the closure of the class of regular languages under union, intersection, and complement.

### Solution

Assume is regular. Then it has a pumping length such that for any string with there exist , , and such that and furthermore

1. For each , ,
2. , and
3. .

Let . Then because of condition 3, where (in particular, consists entirely of ’s). Next, because of condition 2, where , so . However, this contradicts condition 1. Therefore, our assumption that is regular must be false.

§2.1 Context-Free Grammars

### Definition

A **context-free grammar** is a 4-tuple , where

1. is a finite set called the **variables**,
2. is a finite set, disjoint from , called the **terminals**,
3. is a finite set of **rules** of the form , and
4. is the **start variable**.

### Terminology and Notation

The terms **yields**, **derives**, and **language generated by the grammar** and the associated notation are defined as they are for any grammar.

### Example (Sipser Exercise 2.4a)

Give a context-free grammar that generates .

### Solution

### Practice (Sipser Exercise 2.4b)

Give a context-free grammar that generates .

### Solution

### Example (Sipser Exercise 2.6a)

Give a context-free grammar generating the set of strings over the alphabet with more ’s than ’s.

### Solution

First, let denote the number of occurrences of in . Next, let and let be the CFG with rules . Then .

Proof: Suppose . If then . Assume that for some and all it is the case that for all such that we have . Now suppose . Then either (1) , (2) , or (3) . For case (1) we have and where , and by the inductive hypothesis , i.e and . This means that , i.e. . For case (2) we have , and by the inductive hypothesis , i.e. . This means that , i.e. . Case (3) follows similarly. We have shown by the principle of strong mathematical induction on the length of ’s derivation that .

I also claim that , which we can do by induction on where . However, the proof is rather long so we’ll leave it for another day. If you’ll accept that for now, we can conclude that .

Now that we have a grammar that generates , we can use the insights gained from that grammar to design one that generates , as follows:

### Practice (Sipser Exercise 2.6d)

Give a context-free grammar generating .

### Solution

First, I claim the following grammar generates :

How would you prove this claim?

Next, we can build on that grammar as follows: