CS 311: Computational Structures

James Hook

October 23, 2014

8 Chomsky Normal Form and Push Down Automata

8.1 Recall

- Context Free Grammars
- Ambiguity
- Grammar Normal Form (motivation)

8.2 Plan

- Problem Set 3 and Exercise 4 review
- Chomsky Normal Form
- Push Down Automata

8.3 Problem Set 3 and Exercise 4

Any questions?

Midterm exam discussion.

8.4 Chomsky Normal Form

Definition 8.1 *A Grammar $G = (V, T, R, S)$ is in* Chomsky Normal Form *if all productions in R are of one of the following three forms:*

1. *$A \rightarrow BC$ where $A, B, C \in V$ and $B, C \neq S$*
2. *$A \rightarrow a$ where $A \in V$ and $a \in T$*
3. *$S \rightarrow \epsilon$*

In particular, there is no recursion through the sentential symbol S and the only ϵ production allowed is from the sentential symbol.

Theorem 8.2 *Any context-free langauge is generated by a context-free grammar in Chomsky Normal Form.*

Proof of the theorem is done by giving an algorithm to transform an arbitrary CFG $G = (V, T, R, S)$ to an equivalent grammar in CNF.

Steps:

1. Augument with new sentential symbol S_0 and new rule $S_0 \to S$.
2. Eliminate ϵ-rules.
3. Eliminate unit rules.
4. Make all rules of proper form.

8.5 Example

Balanced parens to CNF.

8.6 Push Down Automata

Cartoons.

Formal definition. $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$. Type of δ is $Q \times \Sigma_\epsilon \times \Gamma_\epsilon \to \mathcal{P}(Q \times \Gamma_\epsilon)$.

Definition of acceptance.

Examples.

Variations on PDAs.

Set up parsing machine for equivalence.