CSE396 Lecture Thu 3/5/2015

First, a note on the "Pumping Lemma" and Logic: Version in text Theorem 1.70 (page 78 both eds.)

<u>If</u> A is a regular language, then [$p = |Q|$ for some DFA $M = (Q, \Sigma, \delta, q_0, F)$, $L(M) = A$]

<u>THERE EXISTS</u> an integer $p > 0$ such that $(q_0, s_1, q_1, s_2, q_2, \ldots, q_{p-1}, s_p, q_p, \ldots)$

<u>FOR ALL</u> $s \in A$ with $|s| \geq p$

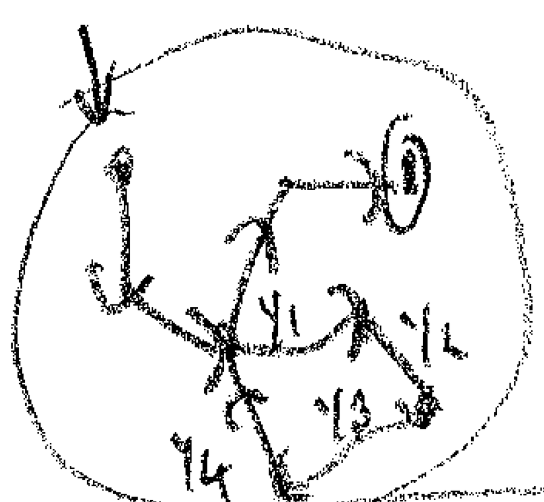
<u>THERE EXISTS</u> a breakdown $s =: x \cdot y \cdot z$ such that

$|xy| \leq p$, $y \neq \varepsilon$, and

<u>FOR ALL</u> $i \geq 0$,

$xy^i z \in A$.

(Note in particular: $i = 0$ gives $xz \in A$, "pumping down".)



$\boxed{\begin{array}{c} Reg(A) \to X \\ \text{contrapositive} \\ \neg X \to \neg Reg(A) \end{array}}$

Contrapositive:

$A_1 = \{w \cdot w^R : w \in \{a,b\}^*\} = \{\text{even-length palindromes}\}$

$A_2 = \{w \cdot w : w \in \{a,b\}^*\} = \{\text{"double words"}\}$.

If A is any language such that

<u>FOR ALL</u> integers $p > 0$

THERE <u>EXISTS</u> $s \in A$ with $|s| \geq p$ such that A_1: Take $s = \underline{a^p bb a^p}$

<u>FOR ALL</u> breakdowns $s =: x \cdot y \cdot z$ with $|xy| \leq p$, $y \neq \varepsilon$,

<u>THERE EXISTS</u> $i \geq 0$ such that Let $x = a^m$ s.t. $m + n \leq p$, $y = a^n$ $n \neq 0$, $z = a^{p-m-n} bb a^p$.

$xy^i z \notin A$. Take $i = 0$: $xz = a^{p-n} bb a^p \notin A_1$ since $n > 0$.

<u>THEN</u> A is not regular.

Proof Script:

<u>Let any</u> $p > 0$ <u>be given</u>. A_2:

$x = a^m$ $y = a^n$, $n \neq 0$, $m + n \leq p$, $z = a^{p-m-n} b a^p b$.

<u>Take</u> $s = \underline{a^p b a^p b}$. Note $s \in A$ and $|s| \geq p$.

$s = a^p b$

<u>Consider any</u> breakdown $s =: x \cdot y \cdot z$ with $|xy| \leq p$, $y \neq \varepsilon$.

Let $x, y \in S$, $x \neq y$, $x = a^m b$ $y = a^n b$

<u>Take</u> $i = \underline{0}$. <u>Then</u> $xy^i z = \underline{a^{p-n} b a^p b} \notin A$ because $n > 0$

Take $z = a^m b$

<u>Hence</u> A is not regular, by the Pumping Lemma.

$xz = a^m b a^m b \in A_2$, $yz = a^n b a^m b \notin A_2$.

Context-Free Grammars:

$A_1 = \{ w \cdot w^R : w \in \{a,b\}^* \}$.

$A_1 = \{ \varepsilon, aa, bb, aaaa, \underset{w=ab}{abba}, \underset{w=ba}{baab}, bbbb, \underset{w=aaa}{aaaaaa}, \underset{aab}{aabbaa}, \dots \}$

A CFG G s.t. $L(G) = A_1$:

S means "I stand for some/any EVENPAL X."

$$S \to aSa \mid bSb \mid \varepsilon \quad \left(= \left\{ \begin{array}{l} S \to aSa, \\ S \to bSb, \\ S \to \varepsilon \end{array} \right\} \right)$$

Derivation:

$$S \underset{S \to aSa}{\Rightarrow} aSa \underset{S \to aSa}{\Rightarrow} aaSaa \underset{S \to bSb}{\Rightarrow} aabSbaa \underset{S \to \varepsilon}{\Rightarrow} aabbaa$$

Inductive Def^n:

Basis: • ε is an "EVENPAL" — Rule: $S \to \varepsilon$.

Induction:
- If X is an EVENPAL, then aXa is also an EVENPAL. — Rule: $S \to aSa$
- If X is an EVENPAL, then bXb is also an EVENPAL. — Rule: $S \to bSb$.

"Catchall Clause": The only EVENPALs are the ones that arise via these three rules.

$aabbaa \in$ EVENPAL
$\Uparrow$
$abba \in$ EVENPAL
$\Uparrow$
$bb \in$ EVENPAL
$\Uparrow$
$\varepsilon \in$ EVENPAL ✓

$A_2 = \{ ww : w \in \{a,b\}^* \}$.

Can we design a CFG G such that G generates exactly those strings in A_2, i.e. $L(G) = A_2$?

Answer: No! ☹ $\quad (\varepsilon,\ a\,a,\ ab\,ab,\ abb\,abb, \dots)$

CFG $\cong$ BNF as taught in CSE 305.

<u>Formal Definition</u>: A context-free grammar (CFG) is a 4-tuple $G = (V, \Sigma, R, S)$ where Σ is the <u>terminal</u> alphabet,

V is a finite set of <u>variables</u>,

$S \in V$ is the <u>start variable</u>, and

R is a finite set of <u>rules</u> of the form $A \to X$

where $A \in V$ and $X \in \underbrace{(V \cup \Sigma)^*}$.

an arbitrary string of variables and/or terminals, including ε allowed.

Before we have $\Sigma = \{a, b\}$.

$V = \{S\}$, $R = \{S \to \varepsilon, S \to aSa, S \to bSb\}$.

"X derives Y in one step:"

Given two strings $X, Y \in (V \cup \Sigma)^*$, we write $X \underset{G}{\Rightarrow} Y$ if X can be broken down as $X = UAW$ such that for some rule $A \to Z$ in R, $Y = UZW$.

$Y = UZW$

$X = UAW$

We write $S \underset{G}{\Rightarrow}^* X$ if there are $Y_1, \dots, Y_k$ s.t. $S \underset{G}{\Rightarrow} Y_1 \underset{G}{\Rightarrow} Y_2 \underset{G}{\Rightarrow} \cdots \underset{G}{\Rightarrow} Y_k \underset{G}{\Rightarrow} X$, and finally $L(G) = \{x \in \Sigma^* : S \underset{G}{\Rightarrow}^* x\}$

Variables can also be written as XML-style <tags>.

$S \to$ <NounPhrase><VerbPhrase> | ...

<NounPhrase> $\to$ <ComplexNoun> | <ComplexNoun><PrepPhrase> | ...

<ComplexNoun> $\to$ <Article><SimpleNoun> | ...

<Article> $\to$ the | a | an

<PrepPhrase> $\to$ <Preposition><ComplexNoun>

<SimpleNoun> $\to$ cat | dog | hat

<Preposition> $\to$ in

$\therefore$ <NounPhrase> $\Rightarrow^*$ the cat in the hat.

[End of Week 6]