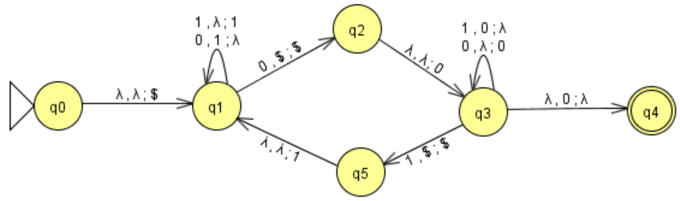
CS 373 Homework 7 Anchu A. Lee
October 19, 2017

1. Construct a pushdown automata that recognizes $\{w \mid w \text{ is an element of } \{0,1\}^* \text{ and } w \text{ has more 0's than 1's }\}$.



2. Convert the following CFG into an equivalent CFG in Chomsky normal form, using the procedure given in Theorem 2.9.

$$A \to BAB \mid B \mid \epsilon$$
$$B \to 00 \mid \epsilon$$

$S \to BC \mid AB \mid BA \mid BB \mid DD \mid \epsilon$
$A \to BC \mid AB \mid BA \mid BB \mid DD$
$B \to DD$
$C \to AB$
$D \to 0$

3. Show that the class of context-free languages is closed under the union operation (construction and proof). The construction should be quite simple.

Proof. Define two context-free languages: $G_1 = (V_1, \Sigma, R_1, S_1)$ and $G_2 = (V_2, \Sigma, R_2, S_2)$ and also the language $G_U = (V_1 \cup V_2 \cup \{S\}, \Sigma, R_1 \cup R_2 \cup \{S \to S_1 \mid S_2\}, S)$ which is the union of G_1 and G_2 as the start variable of G_U points to both start variables of G_1 and G_2. Additionally the rules and variables are shared (assuming the rules and variables are disjoint). After the start variable of G_U, subsequent steps use rules exclusively from G_1 or G_2, not both. therefore all productions of G_U must be in the languages G_1 or G_2. □

4. Show that the class of context-free languages is closed under the concatenation operation (construction and proof). The construction should be quite simple.

Proof. Define two context-free languages: $G_1 = (V_1, \Sigma, R_1, S_1)$ and $G_2 = (V_2, \Sigma, R_2, S_2)$ and also the language $G_C = (V_1 \cup V_2 \cup \{S\}, \Sigma, R_1 \cup R_2 \cup \{S \to S_1 S_2\}, S)$ which is the concatenation of G_1 and G_2 ad the start variable of G_C concatenates both the start variables of G_1 and G_2. So G_C produces words that start with G_1 and end with G_2, thus all productions of G_C must be concatenations of G_1 and G_2. □

5. Show that the class of context-free languages is closed under the star operation (construction and proof). The construction should be quite simple.

Proof. Define the context-free language $G_1 = (V, \Sigma, R, S)$. The star of this language would have to be able to generate Σ or a countably infinite amount of copies. So the start state would have to $S_0 \to \epsilon \mid S_0 S$. Therefore the language $G_S = (V, \Sigma, R \cup \{S_0 \to \epsilon \mid S_0 S\}, S_0)$ generates either ϵ or a sequence of many words in G_1. □

6. Let $M = (Q, \Sigma, \delta, q_0, F)$ be a DFA and define CFG $G = (V, \Sigma, R, S)$ as follows:

   - $V = Q$;
   - For each $q \in Q$ and $a \in \Sigma$, define rule $q \to aq'$ where $q' = \delta(q, a)$;
   - For $q \in F$ define rule $q \to \epsilon$;
   - $S = q_0$.

   Prove $L(M) = L(G)$.

Proof. A language is regular if a DFA accepts it, so $L(M)$ is a regular language. By Corollary 2.32, the language must also be context-free. In order for $L(M) = L(G)$, the construction of G must be a direct translation of a DFA to GFA. G converts the states of a DFA to variables, defines rules that function similarly to the transition functions and uses a rule that moves to ϵ instead of using accept states. This construction successfully translates a DFA into a CFG. $\square$

7. Let $L = \{0^n 1^m 0^n 1^m \mid n, m \geq 0\}$. Show L is not context-free.

Proof. Assume L is context-free. Let p be the pumping length. Let $w = 0^p 1^{p+1} 0^p 1^{p+1}$, which means the options for vxy are 0^p, $0^p 1^{p+1}$, 1^{p+1}, $1^{p+1} 0^p$. Each option is really two options as the string is $0^p 1^{p+1}$ twice. If 0^p is pumped up, then there will be too many characters in one of the zero's. If $0^p 1^{p+1}$ is pumped up, then the left side of the word will be longer than the right side. If 1^{p+1} is pumped up then one of 1's will have too many characters than the other 1's. If $1^{p+1} 0^p$ is pumped up then the middle two 1's and 0's will have more characters than the outer 1's and 0's when they need to be equal. Since every case of vxy is not in L, the language cannot be context-free. $\square$

8. Let $L = \{w \mid w$ is in $\{a, b, c, d\}^*$, with the number of a's = number of b's and the number of c's = the number of d's $\}$. Show L is not context-free.

Proof. Assume L is a context-free language, let p be the pumping length. Let $w = a^p d^{p+1} b^p c^{p+1}$, which means the options for vxy are a^p, $a^p d^{p+1}$, d^{p+1}, $d^{p+1} b^p$, b^p, $b^p c^{p+1}$, c^{p+1}. If a^p is pumped up, then the number of a's do not equal to the number of b's. If b^p is pumped up, then the number of b's do not equal the number of a's. If $b^p c^{p+1}$ is pumped up, then the number of b's do not equal the number of a's or the number of c's do not equal the number of d's. If c^{p+1} is pumped up then the number of c's do not equal the number of d's. If $d^p + 1$ is pumped up then the number of d's do not equal the number of c's. With $a^p d^{p+1}$, the number of a's do not equal the number of b's or the number of d's do not equal the number of c's. If $b^p c^{p+1}$ is pumped up them the number of b's do not equal to the number of b's or the number of c's do not equal the number of d's. Therefore the language cannot be context-free. $\square$

9. Let A and B be languages. We define $A \approx B = \{ab \mid a$ is an element of A and b is an element of B and $|a| > |b|$ $\}$. Show that if A and B are regular languages, then $A \approx B$ is a context-free language.

Proof. $A \approx B$ must be accepted by a PDA if it is context-free. So let C be a PDA that recognizes A and D be a PDA that recognizes B. Then let G be a PDA that recognizes $A \approx B$. The start state of G would simply push a \$ to the stack and transition to the start state of C. Every transition in C also pushes a 0 onto the stack. Each accept state in C ϵ-transitions to the start state of D. Every transition in D also pops a 0 off the stack. If D reaches it's accept state and there are still 0's on the stack, then it transitions into the accept state of G. G then will only accept words a concatenate with b so long as $|a| > |b|$. $\square$

10. Show $L = \{w \mid w$ is an element of $\{a, b, c, d, e, f\}^*$ such that the number of a's + number of b's = number of c's + number of d's = number of e's + number of f's$\}$ is not context-free.

Proof. Assume L is context-free, and p is the pumping length. Choose $w = a^p c^p e^p f^{p+1} d^{p+1} b^{p+1}$, which follows $|w| \geq p$. The string vxy can therefore be:

vxy	Pump up
a^p	$a + b$ would be greater than the rest
$a^p c^p$	$a + b$ or $c + d$ would be greater than $e + f$
c^p	$c + d$ would be greater than the rest
$c^p e^p$	$c + d$ or $e + f$ would be greater than $a + b$
e^p	$e + f$ would be greater than the rest
f^{p+1}	$e + f$ would be greater than the rest
$f^{p+1} d^{p+1}$	$c + d$ or $e + f$ would be greater than $a + b$
d^{p+1}	$c + d$ would be greater than the rest
$d^{p+1} b^{p+1}$	$a + b$ or $c + d$ would be greater than $e + f$
b^{b+1}	$a + b$ would be greater than the rest

No matter which substring used for vxy, it will fail the language requirements. Therefore the language fails the pumping-lemma and cannot be context-free. $\square$