Mathematical Methods for Computer Science 2
Spring 2018

Series 13

Sylvain Julmy

1

a)

$L = \{a^i b^j c^k | i < j < k\}$ is not context-free :
Assume L is context-free, then the Pumping Lemma hold, then for all $\omega \in L$, we have $\omega = uvwxy$ such that $vx \neq \epsilon, |vwx| \leq p$ and $i \geq 0, uv^i wx^i y$.
We pick z as $a^{m+2} b^{m+1} c^m$, where m is the pumping length. z is clearly in L and $|z| \geq m$. We know that $|vwx| \leq m$ and $|vx| > 0$, then we know that vwx must consist of :

1. only a's $\implies vx = a^s$ since the lemma tell us $|vx| > 0$. Now we take $i = 0$, then we have $z' = uv^0 wx^0 y \in L$ but z' is $a^{m+2-s} b^{m+1} c^m$ and since $s > 0$, the pumping lemma don't hold.

2. some a's and some b's $\implies vx = a^s b^t$ such that $s + t > 0$, we take $i = 0$ then z' is in the form $a^{m+2-s} b^{m+1-t} c^m$. If t is positive, then $|b| > |c|$ and its not in the language. If $t = 0$, then its case 1.

3. only b's take $i = 0$, then z' is in the form $a^{m+2} b^{m+1-s} c^m$ and s is positive, which violates the condition that $|b| > |c|$. So $z' \notin L$.

4. some b's and some c's : take $i = 2$ then $z' = a^{m+2} b^{m+1+s} c^{m+t}$, if $s \neq 0$ then $|a| > |b|$ don't hold. If $s \neq 0$ then $t \neq 0$ then $|b| > |c|$ don't hold, so $z' \notin L$.

5. only c's : take $i = 2$, then $z' = a^{m+2} b^{m+1} c^{m+s}$, since s is positive, then $|b| > |c|$ don't hold.

Since all five cases fail to pump, the Pumping Lemma tells us that this language is not context-free.

b)

$\{a^i | i \text{ is a prime}\}$ is not context-free :
Let $v = x^q$ and $y = x^t$, note that the Pumping Lemma requires $q + t > 0$. Let $r = |uxz| = pqt$. Then $|uvrxyrz| = r + rq + rt = r(1 + q + t)$ is divisible by both r and $1 + q + t > 1$ and thus is not prime as long as $r > 1$.
Then there are two unsettled cases: if $r = 0$, $|uv2xy2z| = |v2y2| = 2p$ is not prime. Finally, if $r = 1$, $|uvp + 1xyp + 1pz| = 1 + (p + 1)q + (p + 1)t = 1 + (p + 1)(q + t) = 1 + (p + 1)(p1) = p2$ isn't prime.

2

The language $L = \{0^i 1^j 0^{i+j}\}$ is not regular :
Assume L is regular, let $\omega = 0^s 1^t 0^{t+s}$. Thus $|\omega| = 2(s+t)$, by the pumping lemma, let $\omega = xyz$, where $|xy| \leq n$. Let $x = 0^p$, $y = 0^q$ and $z = 0^r 1^t 0^{p+q+r+t}$ where $p+q+r = s$, $p \neq 0$, $q \neq 0$, $r \neq 0$. Thus $|y| \neq 0$. Let $k = 2$, then $xy^2z = 0^p 0^{2q} 0^r 1^t 0^{p+2q+r+t}$. Then the number of a is $2(p+2q+r) = 2(n+q)$. Hence $xy^2z = 0^p 0^{2q} 0^r 1^t 0^{n+q+t}$, since $q \neq 0$ xy^2z is not of the form $0^i 1^j 0^{i+j}$, thus $xy^2z \notin L$, hence L is not regular.
The language $L = \{0^i 1^j 0^{i+j}\}$ is context-free since the following grammar generate it:

$$A \to 0AB0 \mid \epsilon$$
$$B \to 1B0 \mid \epsilon$$

3

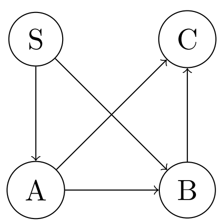
4

If L is context-free, then there is a PDA P accepting it. If M is regular, then there is a DFA F accepting it. The intersection of L and M is the language that accept word that are in L and in M. Any word that are in the intersection are accepted by D but not all the word that are accepted by D, only those accepted by P.
The cross product proof consists of constructing an automaton $P \otimes F$ which contains the mechanics of both P and F, and which accepts only words for which both sides accept. The cross-product automaton is a PDA (and therefore the recognized language is context-free) — intuitively, because the cross product with an n-state DFA consists of taking n copies of P and adding $(q, a, [q])$ arrows between matching states in P where the DFA has a arrows. The result is not a finite automaton in general (not even a non-deterministic one) because the P part relies on the stack and this reliance does not go away in $P \otimes F$ in general.

5

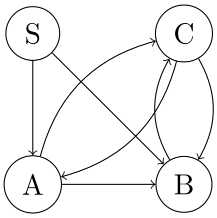
a)

The language generated is finite because there are no cycle in the following graph of possible generation :



b)

The language generated is infinite because we have a cycle between C and S :



The following familly of word is in L :

$$S \to AB \to BCB \to BABB \to BACCB \to BAABCB \to$$
$$BAACCCB \to BAAABCCB \to BAAACCCCB \to$$
$$BAAAABCCCB \to BAAAACCCCCB \to BAAAAABCCCCB \to \dots$$

all the words of the form $ba^nbb^nb = ba^nb^{n+2}$