Theoretical Computer Science (Bridging Course)

Dr. G. D. Tipaldi
F. Boniardi
Winter semester 2014/2015

University of Freiburg
Department of Computer Science

Exercise Sheet 5

Due: 4th December 2014

Exercise 5.1 (Context-free grammars, Pushdown automata)

- Construct a grammar that generates the regular expressions over an alphabet Ξ.

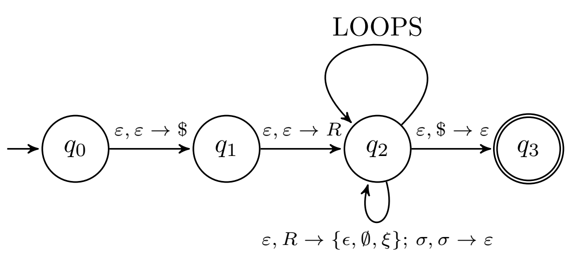
  Solution: According to the definition of regular expression, the following context-free grammar $G := (V, \Sigma, \mathcal{R}, S)$ generates all the regular expression on the alphabet Ξ:

  - $V := \{R\}$.
  - $\Sigma := \Xi \cup \{\epsilon, \emptyset, \cup, \circ, ^*, (,), ^+\}$.
  - $S := R$.
  - The production rules in $\mathcal{R}$ are defined as follows:

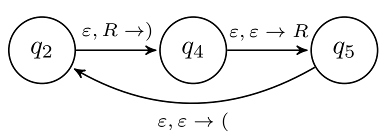
  $$R \to (R) \mid R^+ \mid R^* \mid R \cup R \mid R \circ R,$$
  $$R \to \epsilon \mid \emptyset \mid \xi_1 \mid \cdots \mid \xi_n.$$

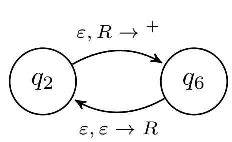
- (*Extra, not mandatory*) Can you design an automation that checks whether a regular expression is well formed?

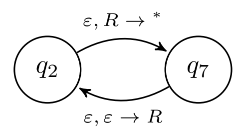
  Solution: In order to check whether a regular expression on the alphabet is well formed we can easily construct an automaton that accept an input if and only if such input is generated by the above grammar. To do so, we can use the PDA reported below. We use ε to distinguish the empty input from the element in V.

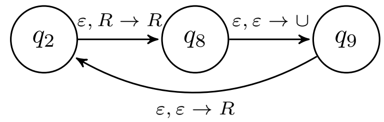


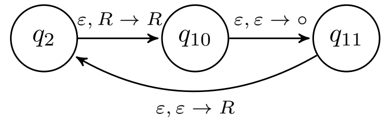
In the diagram above $\varepsilon, R \to \{\epsilon, \emptyset, \xi\}$ and $\sigma, \sigma \to \varepsilon$ should be understood as the set of rules $\{\varepsilon, R \to \epsilon; \varepsilon, R \to \emptyset; \varepsilon, R \to \xi \ (\xi \in \Xi)\}$ and $\{\sigma, \sigma \to \varepsilon \mid \sigma \in \Sigma\}$ respectively. Furthermore the LOOPS label represent the following closed chains:











Exercise 5.2 (Chomsky normal form)

Let $\mathcal{G} = (V, \Sigma, \mathcal{R}, S)$ be a context-free grammar, we call *sentential form* a word $\omega \in (\Sigma \cup V)^*$ so that ω is derived by applying a sequence of production rules in $\mathcal{R}$. Roughly speaking, sentential forms are those sequence of variables and terminal symbols that can be obtained from the production rules of the grammar.

Show that, if $\mathcal{G}$ is a context-free grammar in Chomsky normal form and ω is a non-empty sentential form only composed of non-terminal symbols ($\omega \in V^* \setminus \{\epsilon\}$), then ω can be obtained by applying $r(\omega) = |\omega| - 1$ production rules.

Solution: We can easily prove this using induction on the length of the sentential form. Let's set $n := |\omega|$ and set $V := \{V_1, ..., V_N\}$ to be the variables.

- $\underline{n = 1}$. We observe that, since the grammar is in Chomsky normal form, if $|\omega| = 1$ then $\omega = S$ (just the staring variable, remember that $S \subseteq V \subset V^* \setminus \{\epsilon\}$). Thus no derivation rules are required, that is, $0 = r(\omega) = 1 - 1$.

- $\underline{n - 1 \Rightarrow n}$. Let's take $\omega = V_{i_1} \cdots V_{i_n}$ ($i_1, ..., i_n \in \{1, ..., N\}$) to be a general sentential form with no terminals and whose length is n. If $n \geq 2$, then there exists a "parent" sentential form $\omega' \in V^* \setminus \{\epsilon\}$ so that $|\omega'| = n - 1$ and ω is derived from ω' by applying only one production rule of type $[V_i \to V_j V_k]$. For if it did not, there must exist a sentential form ω^* whose length is $n - k$ ($2 \leq k \leq n - 1$, eventually $\omega^* = S$) so that ω is derived from ω^* by applying only a production rule. In such case, that production rule would be of the form $[V_i \to V_{j_1} \cdots V_{j_{k+1}}]$ and since $k \geq 2$, this would contradict the hypothesis that the grammar is in Chomsky normal form. As a consequence, such ω' exists and from the induction hypothesis $r(\omega') = |\omega'| - 1 = n - 2$. Since ω is obtained from ω' just applying only one production rule, then $r(\omega) = r(\omega') + 1 = n - 1 = |\omega| - 1$.

The statement is proven.

Exercise 5.3 (Pumping Lemma)

Are the following languages context-free?

(a) $L_1 := \{1^n 0^m 1^{nm} \mid n, m \geq 0\}$.

Solution: Let's suppose that the above language is context-free, then, according to the pumping lemma there would exist an integer $p > 0$ so that every word $w \in L_1$ with length at least p could be written as $w = xyzuv$, where x, y, z, u and v satisfy:

(P1) $|yu| \geq 0$.

(P2) $|yzu| \leq p$.

(P3) $xy^k zu^k v \in L_1$ for every $k \geq 0$.

In order to derive a contradiction, let's consider $w = 1^p 0^p 1^{p^2}$, where p is the pumping length and let x, y, z, u and v defined as above. According to (P3), y and u should not contain both 0 and 1, otherwise the pumping procedure would be inconsistent with the ordering of the string (0s and 1s would be mixed up). As a consequence, $y, u \in 0^* \cup 1^*$. It is clear that y and u cannot both lie in 0^*, otherwise only the number of 0s would be affected by pumping the strings and the numerical relation between the number of 0s and 1s would be broken. So only two cases are still feasible:

- $y \in 1^*, u \in 0^*$ or $y \in 0^*, u \in 1^*$. We consider just the former as the latter can be proved with similar arguments. If $y \in 1^*, u \in 0^*$, then the "pumped" string would be

$$xy^k zu^k v = 1^{p-|y|+(k-1)|y|} 0^{p-|u|+(k-1)|u|} 1^{p^2}.$$

  It is easy to see that the only choice of $|y|, |u| \geq$ so that

$$(p - |y| + (k-1)|y|)(p - |u| + (k-1)|u|) = p^2 \quad \forall k \geq 0$$

  is $|y| = |u| = 0$ which contradicts (P1).

- $y, u \in 1^*$. In such case y should be a subset of the fist sequence of 1s and u of the second ones otherwise condition (P2) would not be satisfied. That is

$$\underbrace{11\cdots\underbrace{11\cdots11}_{y}\cdots11}_{p\text{-times}}\underbrace{00\cdots00}_{p\text{-times}}\underbrace{11\cdots\underbrace{11\cdots11}_{u}\cdots11}_{p^2\text{-times}}.$$

  Again, if we pump y and u we have

$$xy^k zu^k v = 1^{p-|y|+(k-1)|y|} 0^p 1^{p^2-|u|+(k-1)|u|},$$

  and the only feasible values for $|y|, |u| \geq 0$ that satisfy the above condition for any choice of $k \geq 0$ are $|y| = |u| = 0$, which contradicts (P1).

Consequently, L_1 is not context free.

(b) $L_2 := \{a^{n^2} \mid n \geq 0\}$

Solution: We can show that this language is not context-free. Indeed, if x, y, z, u and v are substrings satisfying conditions (P1), (P2) and (P3), where the "parent" string is $w := a^{n^2}$ with $|w| \geq p$, then it is easy to see that

$$xy^k zu^k v = a^{n^2+(k-1)(|y|+|u|)}.$$

It is apparent that there exists no $|u|, |y| \in \mathbb{N}_0 \times \mathbb{N}_0 \setminus \{0, 0\}$ and so that $n^2 + (k-1)(|y| + |u|)$ is the square of an integer number for any choice of k.