Computability and Complexity

Instituto de Matemáticas

Exercises

Context-free languages

Exercise 1 : ***Elementary training.***
Let $\Sigma = \{a, b\}$. For each of the following languages, give a context-free grammar that generates it:

1. $\{w \in \Sigma^* \mid w$ contains at least three $b\}$;
2. $\{w \in \Sigma^* \mid w$ starts and ends with the same symbol$\}$;
3. $\{w \in \Sigma^* \mid w$ has odd length$\}$;
4. $\{w \in \Sigma^* \mid w$ has odd length and its middle symbol is $a\}$;
5. $\{w \mid w$ is a palindrome$\}$;
6. $\{w \mid w$ contains as many a as $b\}$.

$\square$

Exercise 2 : ***Ambiguous elementary school.***
Let $\Sigma = \{a, b, c\}$. Give a context-free grammar that generates the language

$$\{a^i b^j c^k \mid i = j \text{ or } j = k;\ i, j, k \geq 0\}.$$

Is your grammar ambiguous? Justify.

$\square$

Exercise 3 : ***Pushdown automatas are not pushovers!.***
For each of the following languages, construct a pushdown automata that recognizes it:

1. $\{w \in \{a, b\}^* \mid w$ has twice as many a than $b\}$;
2. $\{a^i b^j c^k \mid i = j \text{ or } j = k;\ i, j, k > 0\}$;
3. $\{a^m b^n c^{2(m+n)} \mid m, n \geq 0\}$.

$\square$

Exercise 4 : ***Closing exercise.***
Prove that the class of context-free languages is closed under union, concatenation, star and reversal[1].

$\square$

Exercise 5 : ***More pumping!.***
Use the pumping lemma to prove that the following languages are not context-free:

1. $\{a^n b^n a^n b^n \mid n \geq 0\}$;
2. $\{a^n b a^{2n} b a^{3n} \mid n \geq 0\}$;
3. $\{w\#t \mid w, t \in \{a, b\}^*$ and w is a substring of $t\}$.

$\square$

[1] Recall that, given L, its reverse language L^R is the language containing all the words in L in reverse order.

Exercise 6 : ***Problem: doing additions.***
In this exercise, we add a *write* function to a pushdown automata. More formally, we define a pushdown writer-automata as a 7-uple: $A = (Q, \Sigma, \Gamma, W, \delta, q_0, F)$, where W is the writing alphabet, and

$$\delta : Q \times \Sigma_\epsilon \times \Gamma_\epsilon \to \mathcal{P}(Q \times \Gamma_\epsilon \times W_\epsilon)$$

is the transition function. The way such an automata works is the following: it reads a character (which may be ϵ) of the input and may pop the top of the stack, then it changes a state while (maybe) pushing a character on top of the stack and (maybe) writing a character. The characters that are written cannot be modified afterwards. The following figure illustrates how we can draw such an automata:

$$\begin{tikzcd} q \arrow[r, "{a, s \to t, w(\ell)}"] & r \end{tikzcd}$$

Here, when in state q, reading the character a from the input and having s at the top of the stack, we will pop s, push t and write ℓ.
Other functions of the pushdown writer-automata are exactly the same than those of a classical pushdown automata, in particular the acceptance of an input.

1. Draw a pushdown writer-automata that recognizes the language $\{a^n b^n \mid n \geq 0\}$ and writes down the number of a in unary, followed by a #, followed by the number of b in unary. For instance, when reading *aaabbb*, it will write 111#111 and accept the word; and when reading *aabbb*, it will write 11#111 and reject the word.

2. Let w_1 and w_2 be two positive integers written in unary. Draw a pushdown writer-automata that writes their sum in unary. You have to decide the way to represent the two numbers as input. The automata must end its computation in a final state.

3. Let w_1 and w_2 be two positive integers written in binary, such that they have the same number of digits (if they do not, we can simply add 0s at the beginning of the smallest number until it is the case). Draw a pushdown writer-automata that writes their sum in binary. You have to decide the way to represent the two numbers as input. The automata must end its computation in a final state.
<u>Hint:</u> You may have part of the input in reverse order, and write the result in reverse order too.

4. Explain how you could use more stacks in the previous writer-automata to have no reverse order shenanigans.

5. Explain how to construct a pushdown writer-automata to compute the sum of two positive integers written in decimal.

$\square$