Assignment 2 solutions

---

1. For each of the following languages, give a CFG that generates the language:

   (a) $\{w \in \{0,1\}^* : |w|_0 = 2|w|_1\}$.

   (b) $\overline{\text{PAL}}$, where PAL is the language of palindromes (as defined in Lecture 7).

   (c) $\overline{\text{BAL}}$, where BAL is the language of balanced parentheses (also as defined in Lecture 7).

Solution. There are, of course, many CFGs that generate these languages, so your CFGs could differ from these ones and still be correct.

(a) The following CFG generates $\{w \in \{0,1\}^* : |w|_0 = 2|w|_1\}$:

$$S \rightarrow 0\,S\,0\,S\,1\,S \mid 0\,S\,1\,S\,0\,S \mid 1\,S\,0\,S\,0\,S \mid \varepsilon.$$

(b) The following CFG generates $\overline{\text{PAL}}$:

$$\begin{aligned} S &\rightarrow 0\,S\,0 \mid 1\,S\,1 \mid 0\,X\,1 \mid 1\,X\,0 \\ X &\rightarrow 0\,X \mid 1\,X \mid \varepsilon \end{aligned}$$

(c) The following CFG generates $\overline{\text{BAL}}$:

$$\begin{aligned} S &\rightarrow X\,(\,A \mid A\,)\,X \\ A &\rightarrow (\,A\,)\,A \mid \varepsilon \\ X &\rightarrow (\,X \mid)\,X \mid \varepsilon \end{aligned}$$

To understand this CFG, observe first that A generates BAL and X generates all of $\{\,(\,,\,)\,\}^*$. The start variable S therefore generates any string over $\{\,(\,,\,)\,\}$ such that there is either a left-parenthesis with no matching right-parenthesis or a right-parenthesis with no matching left-parenthesis. Every string in $\overline{\text{BAL}}$ has one of these two forms.

2. Let $\Sigma = \{0, 1\}$ and let $A \subseteq \Sigma^*$ be a context-free language. Prove that the following languages are also context-free:

   (a) $B = \{uv : u, v \in \Sigma^* \text{ and there exists } \sigma \in \Sigma \text{ such that } u\sigma v \in A\}$.

   (b) $C = \{u\sigma v : u, v \in \Sigma^*, \sigma \in \Sigma, \text{ and } uv \in A\}$.

   In other words, B is the language of all strings you can obtain by choosing a string from A and removing exactly one symbol from that string, while C is the language of all strings you can obtain by choosing a string from A and inserting exactly one additional symbol from Σ anywhere into that string.

Solution. Because A is a context-free language, we may assume that there exists a CFG G in Chomsky normal form such that $\mathrm{L}(G) = A$. We will refer to this CFG for both parts of the problem.

(a) To prove that B is context-free, it suffices to prove that there exists a CFG H that generates B. We will define H as follows:

- For every variable X appearing in G, the new CFG H will have two variables: X and X_0. The idea is that X will generate exactly those strings in H that it does in G, while X_0 will generate strings that can be obtained by removing a single symbol from any string generated by X.
- For every rule of the form $X \to Y\,Z$ in G, include the following rules in H:

$$\begin{aligned} X &\to Y\,Z \\ X_0 &\to Y_0\,Z \mid Y\,Z_0 \end{aligned}$$

- For every rule of the form $X \to \sigma$ in G, include the following rules in H:

$$\begin{aligned} X &\to \sigma \\ X_0 &\to \varepsilon \end{aligned}$$

- If the rule $S \to \varepsilon$ appears in G, just ignore it.
- Finally, take S_0 to be the start variable of H.

It is evident that $\mathrm{L}(H) = B$, and therefore B is context-free.

(b) To prove that C is context-free, it suffices to prove that there exists a CFG K that generates C. We will define K as follows:

- For every variable X appearing in G, the new CFG K will have two variables: X and X_0. The idea is that X will generate exactly those strings in K that it does in G, while X_0 will generate strings that can be obtained by inserting a single symbol somewhere into any string generated by X.
- For every rule of the form $X \to Y\,Z$ in G, include the following rules in K:

$$\begin{aligned} X &\to Y\,Z \\ X_0 &\to Y_0\,Z \mid Y\,Z_0 \end{aligned}$$

- For every rule of the form $X \rightarrow \sigma$ in G, include the following rules in K:

$$\begin{aligned} X &\rightarrow \sigma \\ X_0 &\rightarrow 0\,\sigma \mid 1\,\sigma \mid \sigma\,0 \mid \sigma\,1 \end{aligned}$$

- If the rule $S \rightarrow \varepsilon$ appears in G, include these rules in K:

$$S_0 \rightarrow 0 \mid 1$$

- Finally, take S_0 to be the start variable of K.

It is evident that $\mathrm{L}(K) = C$, and therefore C is context-free.

3. Let $\Sigma = \{0,1\}$ and let $A \subseteq \Sigma^*$ be a regular language. Prove that the following languages are context-free:

   (a) $B = \{ww^{\mathrm{R}} : w \in A\}$.
   (b) $C = \{uv^{\mathrm{R}} : u, v \in A,\ |u| = |v|\}$.
   (c) $D = \{u1v : u, v \in \Sigma^*,\ |u| = |v|,\ uv \in A\}$.

Solution. Because the language A is regular, there must exist a DFA $M = (Q, \Sigma, \delta, q_0, F)$ such that $\mathrm{L}(M) = A$. We will refer to this DFA for all three parts of the problem.

(a) We will construct at CFG G such that $\mathrm{L}(G) = B$ as follows:

- There will be one variable X_q of G for each state $q \in Q$ of M, and the start variable of G will be X_{q_0}.
- For each state $q \in Q$ and each symbol $\sigma \in \{0,1\}$, include this rule in G:

$$X_q \rightarrow \sigma\, X_r\, \sigma$$

  where $r = \delta(q, \sigma)$.
- For each accept state $q \in F$, include this rule in G:

$$X_q \rightarrow \varepsilon$$

It holds that $\mathrm{L}(G) = B$, with the reasoning being essentially the same as in the second proof of Theorem 9.2 in Lecture 9. The language B is therefore context-free.

(b) We will construct at CFG H such that $\mathrm{L}(H) = C$ as follows:

- There will be one variable $X_{p,q}$ of H for each pair of states $(p, q) \in Q \times Q$ of M, and the start variable of H will be X_{q_0,q_0}.
- For each pair of states $(p, q) \in Q \times Q$ and each pair of symbols $(\sigma, \tau) \in \{0,1\} \times \{0,1\}$, include this rule in H:

$$X_{p,q} \rightarrow \sigma\, X_{r,s}\, \tau$$

  where $r = \delta(p, \sigma)$ and $s = \delta(q, \tau)$.
- For each pair of accept states $(p, q) \in F \times F$, include this rule in H:

$$X_{p,q} \rightarrow \varepsilon$$

It holds that $\mathrm{L}(H) = C$. Again the reasoning is similar to the second proof of Theorem 9.2 in Lecture 9, except that we are running two simulations of M, one that generates a string $u \in A$ on the left and one that generates a string v^{R}, for $v \in A$, on the right (with the two strings necessarily having the same length). The language C is therefore context-free.

(c) We will construct at CFG K such that $\mathrm{L}(K) = D$ as follows:

- There will be one variable $X_{p,q}$ of K for each pair of states $(p, q) \in Q \times Q$ of M, as well as one additional variable S, which we take as the start variable of K.
- For every choice of states $p, q, r, s \in Q$ and symbols $\sigma, \tau \in \Sigma$ that satisfy $\delta(p, \sigma) = r$ and $\delta(s, \tau) = q$, include this rule in K:

$$X_{p,q} \to \sigma\, X_{r,s}\, \tau.$$

- For every state $p \in Q$, include this rule in K:

$$X_{p,p} \to 1$$

- For every accept state $q \in F$, include this rule in K:

$$S \to X_{q_0,q}$$

It holds that $\mathrm{L}(K) = D$. We are once again using a methodology similar to the second proof of Theorem 9.2 Lecture 9, building on the previous answer. This time we are again running two simulations of M, with the one that generates the left-hand side of the string running M as usual and the one generating the right-hand side of the string running M in reverse. Every derivation begins with $S \Rightarrow X_{q_0,q}$ for some accept state q, rules of the form $X_{p,q} \to \sigma\, X_{r,s}\, \tau$ are performed, which simulates M running forward on σ and backward on τ, and the derivation ends with an application of a rule $X_{p,p} \to 1$. This puts the 1 in the middle of the string, and is only possible when the two simulations are on the same state—which guarantees that M accepts uv whenever $u1v$ is generated. The language D is therefore context-free.

4. Prove that the following language is not context-free:

$$A = \{w \in \{0,1,2\}^* : |w|_0 \leq |w|_1 \leq |w|_2\}.$$

Solution. Assume toward contradiction that A is context-free. By the pumping lemma for context-free languages, there exists a pumping length n for A.

Let $w = 0^n 1^n 2^n$. It holds that $w \in A$ and $|w| = 3n \geq n$, so it is possible to write $w = uvxyz$ for strings $u, v, x, y, z \in \{0,1,2\}^*$ such that (i) $vy \neq \varepsilon$, (ii) $|vxy| \leq n$, and (iii) $uv^i xy^i z \in A$ for all $i \in \mathbb{N}$.

Because $|vxy| \leq n$, it is not possible that the string vy contains both the symbol 0 and the symbol 2, as these symbols are separated by n occurrences of the symbol 1 in w. There are two cases to be considered:

Case 1: the string vy does not contain the symbol 0. In this case, one may consider $i = 0$. As vy cannot be the empty string, it follows that either

$$\left|uv^0xy^0z\right|_0 > \left|uv^0xy^0z\right|_1 \quad \text{or} \quad \left|uv^0xy^0z\right|_0 > \left|uv^0xy^0z\right|_2$$

and therefore $uv^0xy^0z \notin A$. This contradicts the requirement that $uv^ixy^iz \in A$ for all $i \in \mathbb{N}$.

Case 2: the string vy does not contain the symbol 2. In this case, one may consider $i = 2$. As vy cannot be the empty string, it follows that either

$$\left|uv^2xy^2z\right|_0 > \left|uv^2xy^2z\right|_2 \quad \text{or} \quad \left|uv^2xy^2z\right|_1 > \left|uv^2xy^2z\right|_2$$

and therefore $uv^2xy^2z \notin A$. Again, this contradicts the requirement that $uv^ixy^iz \in A$ for all $i \in \mathbb{N}$.

We have obtained a contradiction in both cases. It therefore follows that A is not context-free.