Exercise set 5 (Pushdown automata)

1. Design PDA's to accept the following languages over $\Sigma := \{a, b\}$:

(a) $\{\alpha \in \Sigma^* \mid \alpha$ contains more a's than b's$\}$.

(b) $\{\alpha \in \Sigma^* \mid \alpha$ contains more b's than a's$\}$.

(c) $\{\alpha \in \Sigma^* \mid \alpha$ contains unequal numbers of a's and b's$\}$.

(d) $\{\alpha \in \Sigma^* \mid \alpha$ contains twice as many a's as b's$\}$.

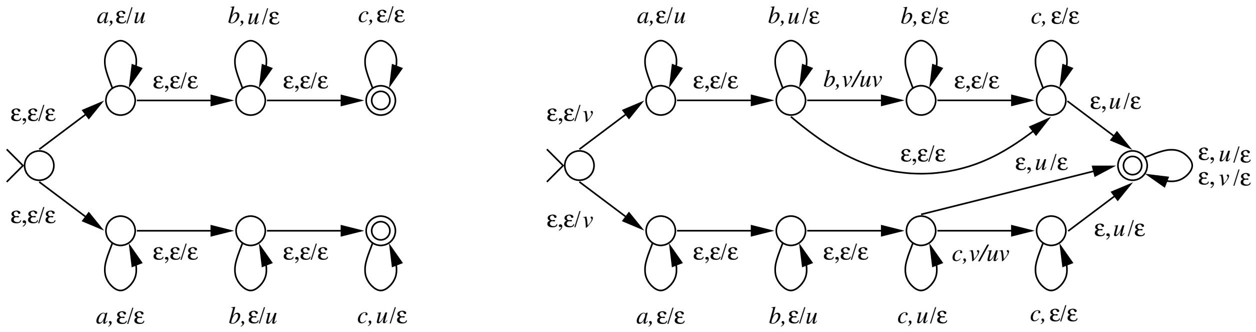
(e) $\{a^k b^k \mid k \in \mathbb{Z}_+\}$.

(f) $\{a^k b^l a^k \mid k, l \in \mathbb{Z}_+\}$.

(g) $\{a^k b^l \mid k, l \in \mathbb{Z}_+, k \leqslant 2l\}$.

(h) $\{\alpha \in \Sigma^* \mid \alpha$ is *not* a palindrome$\}$.

*** 2.** Give informal descriptions (i.e., descriptions in mathematical English) of the languages of the following PDA's. In both the cases assume that $\Sigma := \{a, b, c\}$. For the former PDA take $\Gamma := \{u\}$ and for the second take $\Gamma := \{u, v\}$.



3. **(a)** Design a PDA that accepts the strings over $\Sigma := \{(,)\}$ having balanced parentheses.

(b) Convert the CFG's of Exercise 7 of Exercise set 4 to equivalent PDA's using the CFG-to-PDA conversion procedure described in the lectures.

4. Let $M = (Q, \Sigma, \Gamma, \delta, s, F)$ be a PDA. We define the following languages associated to M:

$$\mathcal{L}(M) := \{\alpha \in \Sigma^* \mid (s, \alpha, \epsilon) \vdash^* (f, \epsilon, \epsilon) \text{ for some } f \in F\} \quad \text{(Language of } M\text{)}$$
$$\mathcal{L}_e(M) := \{\alpha \in \Sigma^* \mid (s, \alpha, \epsilon) \vdash^* (q, \epsilon, \epsilon) \text{ for any } q \in Q\} \quad \text{(Language accepted by empty stack)}$$
$$\mathcal{L}_f(M) := \{\alpha \in \Sigma^* \mid (s, \alpha, \epsilon) \vdash^* (f, \epsilon, \beta) \text{ for some } f \in F \text{ and any } \beta \in \Gamma^*\} \quad \text{(Language accepted by final state)}$$

Clearly $\mathcal{L}(M) \subseteq \mathcal{L}_e(M)$ and $\mathcal{L}(M) \subseteq \mathcal{L}_f(M)$, where these containments may be proper. We also have $\mathcal{L}(M) = \mathcal{L}_e(M) \cap \mathcal{L}_f(M)$. Prove the following assertions that imply that the notions of $\mathcal{L}$, $\mathcal{L}_e$ and $\mathcal{L}_f$ are essentially equivalent.

(a) There exists a PDA M_1 with $\mathcal{L}(M_1) = \mathcal{L}_e(M)$.

(b) There exists a PDA M_2 with $\mathcal{L}_e(M_2) = \mathcal{L}(M)$, provided that $\epsilon \in \mathcal{L}(M)$.

(c) There exists a PDA M_3 with $\mathcal{L}(M_3) = \mathcal{L}_f(M)$.

(d) There exists a PDA M_4 with $\mathcal{L}_f(M_4) = \mathcal{L}(M)$.

5. The condition $\epsilon \in \mathcal{L}(M)$ in Exercise 4(b) is too restrictive. In view of this some authors provide a modified definition of a PDA in which there is a distinguished start symbol $z_0 \in \Gamma$. The modified PDA is defined as a 7-tuple $M = (Q, \Sigma, \Gamma, z_0, \delta, s, F)$. When the PDA starts its operation, its stack is assumed to contain only a single occurrence of the special symbol z_0 and nothing else. (Recall that by our definition a PDA starts its operation with an initially empty stack.) Other symbols in the definition of a modified PDA bear the same meanings as those for our unmodified PDA. Also the criterion for the acceptability of a string (empty stack *and* final state) by a modified PDA is the same as that by an unmodified PDA.

Prove the following assertions:

(a) Given a modified PDA M, there exists an unmodified PDA M' with $\mathcal{L}(M') = \mathcal{L}(M)$.

(b) Given an unmodified PDA M, there exists a modified PDA M' with $\mathcal{L}(M') = \mathcal{L}(M)$.

6. Let M be a PDA. Show that:

(a) There exists a PDA M' with only two stack symbols such that $\mathcal{L}(M') = \mathcal{L}(M)$.

(b) There exists a PDA M'' with only two states such that $\mathcal{L}(M'') = \mathcal{L}(M)$.

* **7.** Let $L \subseteq \{a, b, c\}^*$ and $M := \{b, c, d\}^*$ be two languages. Consider the language $L \oplus M$ over $\{a, b, c, d\}$ which contains all (and only) the strings $\alpha \in \{a, b, c, d\}^*$ with the property that if all a's are deleted from α, we get a string in M and if all d's are deleted from α, we get a string in L. Prove or disprove:

(a) If L and M are regular, then so is $L \oplus M$.

(b) If L and M are context-free, then so is $L \oplus M$.

8. Let L be a context-free language and R a regular language over some alphabet Σ. Prove that:

* **(a)** $L \cap R$ is context-free.

(b) $L \setminus R$ is context-free.

(c) $R \setminus L$ need not be context-free.

9. Prove the following stronger version of the pumping lemma for CFL's:

Let L be a context-free language over an alphabet Σ. Then there exists a constant $n \in \mathbb{N}$ (depending only on L) such that for any $\alpha \in L$ with $|\alpha| \geqslant n$ there exist $\beta_1, \beta_2, \beta_3, \beta_4, \beta_5 \in \Sigma^*$ with the following properties:
(1) $\alpha = \beta_1\beta_2\beta_3\beta_4\beta_5$.
(2) $|\beta_2\beta_4| \geqslant 1$.
(3) $|\beta_2\beta_3\beta_4| \leqslant n$.
(4) $\beta_1\beta_2^k\beta_3\beta_4^k\beta_5 \in L$ for every $k \in \mathbb{Z}_+$.

(**Hint:** Note that condition (3) is extra here and tells that the "middle portion" of α is not too long. For the proof you may mimic the proof for the weaker version, carried out in the lectures. The case $\phi(G) \leqslant 1$ is very uninteresting (Why?). So assume that $\phi(G) > 1$ and take $n := \phi(G)^{|V|+1}$. Also choose the repeated nonterminal A in the long path of T a bit carefully. That's all.)

10. Let $\Sigma := \{a, b\}$ and $L := \{\alpha\alpha \mid \alpha \in \Sigma^*\}$.

* **(a)** Show that L is not context-free.

** **(b)** Show that the complement $\bar{L}$ of L is context-free.