

Note: Use L^AT_EX or Word to typeset your solutions, or write them legibly by hand and submit the scan. You can use the source code of this file as a template.

Problem 1 (PDA with non-empty stack). Formally prove the claim we made in class, that a standard PDA is equivalent to a PDA whose accepting condition is purely based on the state.

For a PDA $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$, let $L'(M)$ be the set of inputs w that M could reach an accepting state $q \in F$ after reading w , but not necessarily having an empty stack on the state q .

1. (10 pts). Let M be a PDA. Construct a PDA M' such that $L(M') = L'(M)$.

Hint. How to empty the stack of M ?

2. (10 pts). Let M be a PDA. Construct a PDA M' such that $L'(M') = L(M)$.

Hint. How to make sure that the stack of M is empty?

Problem 2 (Determine context-freeness). Determine whether the following languages are context-free or not, and justify your answers.

1. (5 pts). $\{a^m b^n \mid m \neq n\}$.
2. (5 pts). $\{a^m b^n c^p \mid p = m + n\}$.
3. (10 pts). $\{a^m b^n c^p \mid p = mn\}$.
4. (10 pts). $\{w \in \{a, b\}^* \mid w \text{ contains strictly more } a\text{'s than } b\text{'s}\}$.
5. (10 pts). $\{x \# y \# z \mid x, y, z \in \{0, 1\}^*, x + y = z \text{ as binary numbers}\}$.
6. (Bonus, 10 pts). $\{a^m b^n c^p \mid m \neq n, n \neq p, m \neq p\}$.
7. (Bonus, 10 pts). $\{wtw^R \mid w, t \in \{a, b\}^*, |w| = |t|\}$.

Hint. The closure property under intersection with regular languages may come in handy when proving non-context-freeness.

Problem 3 (Balanced concatenation). For two languages A and B , define

$$A \diamond B = \{xy \mid x \in A, y \in B, |x| = |y|\}.$$

1. (10 pts). Prove that when A and B are both regular, $A \diamond B$ is context-free.
2. (10 pts). Show that when A is context-free and B is regular, $A \diamond B$ may not be context-free.

Problem 4 (Two-stack PDA). (20 pts). A two-stack pushdown automaton (2PDA) $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$ is defined similarly as a pushdown automaton, but with two stacks instead of one. Formally, the transition function δ has the form

$$\delta : Q \times (\Sigma \cup \{\varepsilon\}) \times (\Gamma \cup \{\varepsilon\}) \times (\Gamma \cup \{\varepsilon\}) \rightarrow 2^{Q \times \Gamma^* \times \Gamma^*},$$

and $(q', t_1, t_2) \in \delta(q, a, s_1, s_2)$ means that there is a transition from state q to state q' that reads the input symbol a , pops s_1 while pushing t_1 in the first stack, and at the same time pops s_2 while pushing t_2 in the second stack. The 2PDA accepts when it reaches an accepting state and both stacks are empty.

Prove that two-stack PDAs can simulate single-tape Turing machines: Given a single-tape Turing machine M , describe the 2PDA M' such that $L(M') = L(M)$.

Hint. Connect the two stacks in a certain way to simulate the tape and head movement.