School of Computing and Information Systems
COMP30026 Models of Computation Problem Set 10

4 – 8 October 2021

Content: pushdown automata, context-free grammars, closure results for context-free languages

P10.1 Construct a push-down automaton for $\{w \mid \text{the length of } w \text{ is odd and its middle symbol is } \mathtt{0}\}$, over the alphabet $\Sigma = \{\mathtt{0}, \mathtt{1}\}$. What's the minimum number of states you can achieve?

P10.2 Take any context-free grammar in any problem set or tutorial, and construct a pushdown automata which recognises the language of that grammar. Try using the CFG to PDA conversion from the lectures, otherwise you can try to understand what the language is, and construct a PDA from scratch. Repeat this a few times for practice.

P10.3 We have seen that the set of context-free languages is not closed under intersection. However, it *is* closed under intersection with regular languages. That is, if L is context-free and R is regular then $L \cap R$ is context-free.

We can show this if we can show how to construct a push-down automaton P' for $L \cap R$ from a push-down automaton P for L and a DFA D for R. The idea is that we can do something similar to what we did in Exercise 65 when we built 'product automata', that is, DFAs for languages $R_1 \cap R_2$ where R_1 and R_2 were regular languages. If P has state set Q_P and D has state set Q $'$

More precisely, let $\delta_D, q_D, F_D)$. Recall the types of the transit

$$\delta_P : (Q_P \times \Sigma_\epsilon \times \Gamma_\epsilon)$$
$$\delta_D : (Q_D \times \Sigma) \to Q$$

We construct P' with the following components: $(q_P, q_D), F_P \times F_D)$. Discuss how P' can be constructed from P and D. Then give a formal definition of δ, the transition function for P'.

P10.4 Give a context-free grammar for $\{\mathtt{a}^i\mathtt{b}^j\mathtt{c}^k \mid i = j \vee j = k \text{ where } i, j, k \geq 0\}$. Is your grammar ambiguous? Why or why not?

P10.5 Consider the context-free grammar $G = (\{S, A, B\}, \{\mathtt{a}, \mathtt{b}\}, R, S)$ with rules R:

$$\begin{array}{rcl} S & \to & A\ B\ A \\ A & \to & \mathtt{a}\ A \mid \epsilon \\ B & \to & \mathtt{b}\ B \mid \epsilon \end{array}$$

(a) Show that G is ambiguous.

(b) The language generated by G is regular; give a regular expression for $L(G)$.

(c) Give an unambiguous context-free grammar, equivalent to G. Hint: As an intermediate step, you may want to build a DFA for $L(G)$.

P10.6 Give a context-free grammar for the language recognised by this DFA: