COMP0003 Theory of Computation
Exercises III: Context-free languages

Yuzuko Nakamura

3 & 5 December 2019

1 Context-free grammars

Exercise 1. Create a CFG that generates sets of properly nested parentheses (), curly braces {}, and square brackets []. To be more specific, (1) all braces must have a match with the opening brace in a pair appearing before the closing brace, and (2) if a pair of braces of one type contains a brace of another type, it must also contain that brace's pair. For example, your grammar should be able to generate strings like "([]{})", "{{}{}}" and "()([()]())" but not strings like "())(", "({)}" or "[[()(])]"

Exercise 2. Describe the language generated by the following grammar:

S $\to$ 1A | 0A
A $\to$ ABB | ε
B $\to$ 0 | 1

Exercise 3. Create a CFG that generates the language $\{w \mid w$ contains more `0`'s than `1`'s$\}$

2 Pushdown automata

Exercise 4. Create a PDA that recognizes the language in Exercise 1.

3 Pumping lemma for CFLs

Exercise 5. Prove that the language $L = \{a^i b^j c^i d^j \mid i, j \geq 0\}$ is not a CFL. ($\Sigma = \{a, b, c, d\}$)