Computation Theory (COMP 170), Fall 2020
Recitation 06

---

[1] CFG $\rightarrow$ PDA

Use the procedure of Lemma 2.21 (Sipser, 3rd ed., pp. 117–120) to construct a PDA for the following grammar $G = (\{E, T, F\}, \{a, *, +, (,)\}, R, E)$ with production rules R as follows:

$$
\begin{aligned}
E &\rightarrow E + T \mid T \\
T &\rightarrow T * F \mid F \\
F &\rightarrow (E) \mid a
\end{aligned}
$$

Note: in this grammar, there are no spaces between symbols.

Note: In lecture, we allowed pushing multiple symbols on the stack at a time, whereas Sipser restricts pushing to one symbol at a time. Your solution can be either.

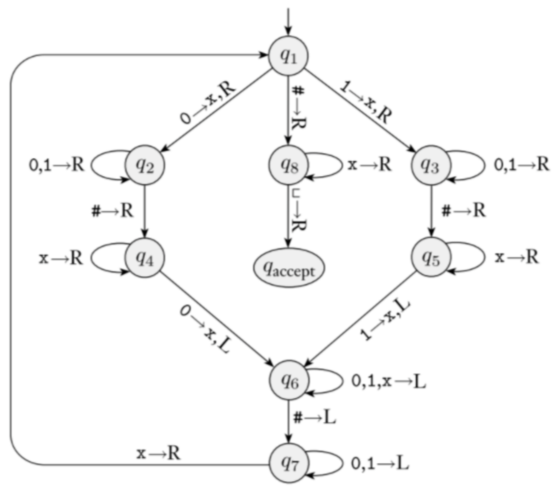
[2] PDA Computation

What would the machine from the previous question do, given the string $a + a \in L(G)$?

[3] Turing Machines

Sipser (3rd ed., p. 173) gives the Turing machine M_1, which accepts the language:

$$B = \{w\#w \mid w \in \{0,1\}^\star\}$$



a. For clarity, the reject state of this machine is implicit. How do we decide when to transition to the reject state?

b. What are the configurations entered by M1 on the following inputs?

`1#0`   `0#0`