Tutorial 3
Theory of Computation

呂紹甲
(Lu, Shao-chia)
11/23
Homework 3

- We have 5 questions this time:

  Q1: Very Easy
  Q2: Easy
  Q3: Easy
  Q4: Moderate
  Q5: Easy/Moderate
  Q6: (Further studies): Hard to Think
1. Let $k$-PDA be a pushdown automaton that has $k$ stacks
   - Thus a 0-PDA is an NFA and a 1-PDA is a conventional PDA.

- We already know that 1-PDAs are more powerful than 0-PDAs (why?)
(a) Show that some language can be recognized by a 2-PDA but not a 1-PDA.

Hint:
Find a (simple) non-CFL that can be recognized by 2-PDA

- Conclude that 2-PDAs are more powerful than 1-PDAs (How?)
(b) (Further studies)

Show that if $L$ can be recognized by a 3-PDA, $L$ can be recognized by some 2-PDA

(Hint: use some kind of encoding)

⇒ If the above is true, we can conclude that 2-PDAs are as powerful as 3-PDAs (why?)
2. Show that:

L is decidable

if and only if

some enumerator enumerates L in

**lexicographic order**
Homework 3

3. Let $S = \{ <M> | M$ is a DFA that accepts $w$ whenever it accepts $w^R \}$

Show that $S$ is decidable.

Hint:
If $M$ recognizes $L$, can we find an NFA $N$ that recognizes $L'$, where $L' = \{ w^R | w$ is in $L \}$?

If $M$ and $N$ are found. Can we decide if $M$ is in $S$?
4. Let $PAL_{DFA} = \{ <M> | M \text{ is a DFA that accepts some palindrome} \}$

Show that $PAL_{DFA}$ is decidable.

Hint:

(i) Fact: $\text{CFL} \cap \text{Reg} \to \text{CFL}$ (Prob 2.18)

(ii) Prob 4.23 shows how to prove a similar language is decidable
5. Suppose that we have a decider $D$ such that $D$ decides if the language of a CFG is infinite. That is,

$D$ is a decider for the language:

$INFINITE_{CFG} = \{ <G> | G$ is a CFG and $L(G)$ is infinite $\}$. 

*BTW, does $D$ exist?*
By using \( D \) or otherwise, show that
\[
C_{CFG} = \{ <G, k> \mid G \text{ is a CFG and } L(G) \text{ contains exactly } k \text{ strings where } k \geq 0 \text{ or } k = \infty \}
\]
is decidable.

Hint:
Let \( p \) be the pumping length of \( G \).
If \( L(G) \) is finite, \( L(G) \) cannot have any string longer than \( p \) (why?)
6. (Further studies)

Prove that:

C is Turing-recognizable if and only if

a decidable language $D$ exists such that $C = \{x \mid \exists y (<x, y> \in D) \}$. 