# CS 4510: Automata and Complexity
**Spring 2015**

Home work 4 // Due: Friday, March 6, 2015

1. (15 points) If $A$ and $B$ are languages, define

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

Show that if $A$ and $B$ are regular languages, then $A \diamond B$ is a context-free language.

**Solution:**

**Idea:** Let $M_1$ be a DFA that acepts $A$ and $M_2$ be a DFA that accepts $B$. A PDA $P$ on input $w$ simulates $M_1$ ont he first part pusing a symbol for every input symobol read so far. It then starts simulating $M_2$ on the second part of $w$ popping a symbol for every input read. The machine $P$ accepts iff both $M_1$ and $M_2$ accepts and the stack is emepty.

2. Assume the following fact as given:

**FACT:** Every CFL without $\epsilon$ can be generated by a CFG in which each rule is in one of the three forms below:

$$\begin{aligned} A &\rightarrow a \\ A &\rightarrow aB \\ A &\rightarrow aBC \end{aligned}$$

where $A, B$ are variables and $a$ is an alphabet symbol.

(15 points) Show that every CFL without $\epsilon$ can be generated by a CFG such that no consecutive symbols on the right-hand side of rules are variables.

**Solution:** (Hopcroft and Ullman, "Introduction to Automata Theory, Languages, and Computation", Exercise 4.16).

Let $G = (V, \Sigma, R, S)$ be a CFG in a normal form stated in the given FACT. The following steps converts $G$ into a grammar $G'$ as required:

- Replace each rule of the form $A\ rightarrow\ aBC$ by $A \rightarrow a[BC]$, where $[BC]$ is a new variable.
- For each new variable $[BC]$, $B$-rule $B \rightarrow \alpha$, and $C$-rule $C \rightarrow \beta$, add the rule $[BC] \rightarrow \alpha\beta$. (Note that $\alpha$ and $\beta$ are single alphabet symbols or of the form $aD$, where $a$ is an alphabet symbol and $D$ is a new or old variable.)

3. (15 points) Let $G$ be a CFG in Chomsky normal form that contains $b$ variables. Show that if $G$ generates some string using a derivation with at least $2^b$ steps, $L(G)$ is infinite.

You can assume as given the following fact:

**FACT:** If $G$ is a context-free grammar in Chomsky normal form, then for any string $w \in L(G)$ of length $n \geq 1$, exactly $2n - 1$ steps are required for any derivation of $w$.

**Solution:** Let $w \in L(G)$ be the string which requires at least $2^b$ derivation steps. Then by the given fact, its length is more than $2^{b-1}$. We claim that in the derivation tree for $w$, there is some path of length at least $b+1$. (Here length of the path is the number of edges in it.) If all paths have length at most $b$ in the derivation tree, then the length of $w$ can be at most $2^{b-1}$. This is true because the grammar is in Chomsky normal form and hence the any node has at most two children. Moreover the leaves have to come from nodes which have exactly one child, and the number of leaves is precisely the length of the string $w$. This is a contradiction to the fact that the length of $w$ is more than $2^{b-1}$. Thus there is some path of length at least $b+1$ in the derivation tree. That is, this path has at least $b+2$ nodes. Since the last node in the path is in $\Sigma$ and there are at most $b$ variables, some variable must repeat in that path. Now similar to the proof of Pumping Lemma for CFG's, we have that $L(G)$ has infinite cardinality.

4. (15 points) Show that the following language is not context-free using the pumping lemma:

$$\{t_1\#t_2\#\cdots\#t_k \mid k \geq 2, \text{ each } t_i \in \{a,b\}^*, \text{ and } t_i = t_j \text{ for some } i \neq j\}$$

**Solution:** Given a constant $p$, choose the string $s = a^p b^p \# a^p b^p$.

5. Define a *Constrained Two-Headed* (CTH) machine as follows.

   An CTH machine is a DTM with one tape that is unbounded on one side and that has two heads FH and BH.

   - The head FH is a read-only head and the head BH is a write-only head.
   - Both heads can only move right.
   - If a symbol on the tape under the read head FH is read, the head FH moves right by one position.
   - If a symbol is written on the tape under the write head BH, the head BH moves right by one position.
   - In each step, the machine $M$ being in a state and either reading the symbol on the tape under the head FH or not reading the symbol can: (a) change state, and (b) either write a symbol on the tape or not write a symbol.
   - The string on the tape in positions $i$ through $j$ is said to be *current* if FH is on position $i$ and BH is on position $j+1$.

   The following is an example of how a CTH machine computes.

   **Example:** Let the current region on the tape be 100°0110$. (That is the head FH is pointing to 1 and the head BH is pointing to the blank symbol ⊔ after the $ symbol on the tape. The alphabet consists of $0, 1$ and marked 0.)

   The CTH machine writes the string 1010°110$ after this string on the tape and make the new string the current region as follows:

```
WHILE the symbol under the FH head is not $ DO
     IF this symbol is not marked 0 THEN write it on the tape
     ELSE write the symbol 1 on the tape
          Read the next symbol (which is a 0) and write a marked 0 on the tape

Write $ on the tape
```

   That is, the tape is modified so that is has the string 100°0110$1010°110$ with the string 1010°110$ as its current string.

   (a) (10 points) Let the current region on the tape be 100°0110$. (That is the head FH is pointing to 1 and the head BH is pointing to the blank symbol ⊔ after the $ symbol on the tape.)

   Show how to write the string 10°10110$ after this string on the tape and make the new string the current region. (That is, modify the tape so that is has the string 100°0110$10°10110$ with the string 10°10110$ as its current string.)

   (b) (5 points) Write a formal definition of a CTH machine.

   **Solution:** The machine: $N = (Q, \Sigma, \Gamma, \delta, q_s, q_a)$, where
   $\delta : Q \times (\Gamma \cup \{\epsilon\}) \rightarrow Q \times (\Gamma \cup \{\epsilon\})$.
   $\delta(\text{state, input symbol or } \epsilon) = (\text{state, output symbol or } \epsilon)$.

6. Given any TM $M$, construct a 2-PDA $P$ that accepts the same language as $M$ using the following steps. Let $S1$ and $S2$ denote the two stacks. Let $w_1 w_2 \cdots w_m$ be the input.

   Maintain the invariant that at any time, the contents of $S1$ are the contents of the tape of $M$ to the left of the input head and the contents of $S2$ are the contents of the tape of $M$ from the head location to the right end of the tape (the right-most non-blank cell).

   In a tape we can go to a arbitrary cell and perform some operations on it. But a normal PDA can not read, say the fifth character from the bottom of the stack and modify it without altering the contents of the stack above it.

   (a) (3 points) Show how to store the input onto $S2$ such that the first character from the input is at the top.

   **Solution:** Push the input onto $S1$; Pop the contents of $S1$ and push it onto $S2$.

   (b) Let the current state of $M$ be $q$. Let $M$ on seeing an $a$ at state $q$ write a $b$, move left and enter state $q'$.

   (5 points) How do you simulate this move ?

   **Solution:** pop $a$ from $S2$; push $b$ onto $S2$; pop the top symbol of $S1$ and push it onto $S2$.

(c) Let $M$ on seeing an $a$ at state $q$ write a $b$, move right and enter state $q'$.
(5 points) How do you simulate this move ?
**Solution:** push $b$ onto $S1$; pop $a$ from $S2$.

(d) (2 point) When does $P$ accept? **Solution:** when $M$ accepts.