Name: Rollno:

CSE340: Theory of Computation (Homework Assignment 3)

Due Date: 6th October, 2021 11.59PM

Total Number of Pages: 3 Total Points 50

Question 1. (15 points) (a) $\Sigma = \{0, 1, \$\}$. Language $L = \{a\$b \mid a, b \in \{0, 1\}^*, \#a = \#b\}$.
From the start state, on seeing $0, 1$, move to state a (keeps track of a), on seeing w move to the final state. In state a, on seeing $0/1$, push to the stack and loop. On seeing w move to state b. In b, on seeing $0/1$ pop from the stack and loop. If the stack becomes empty or the input symbols, move to the final state. In the final state, on seeing $0/1$ or non empty stack move to a dead state and loop.

(b) $\Sigma = \{0, 1, \$\}$. $L = \{a\$b \mid a, b \in \{0, 1\}^*, a = b\}$

(c) Let C be the language $\{ww|w \in \{0, 1\}^*\}$. C accepts only repeated words. Let's use the pumping lemma to prove that C is not context-free. For the sake of contradiction, assume C is context-free. Let p be the pumping length given by the lemma. Pick a string $s = 0^p1^p0^p1^p$. Clearly $s \in C$ and and $|s| \geq p$. So s can be divided into $uvxyz$ such that all three conditions are satisfied. Since $|vxy| \leq p$, only the following cases can occur:

- Case 1: vxy belongs completely within the first $2p$ characters. Since either v or y is non-empty, uv^2xy^2z cannot be of the form ww.
- Case 2: vxy belongs completely within the last $2p$ characters. Identical to Case 1, uv^2xy^2z cannot be of the form ww.
- Case 3: vxy belongs completely within the middle $2p$ characters. Now, the string uv^0xy^0z has the form $0^p1^q0^r1^p$. Since v and y have been pumped down and at least one of them is nonempty, at least one of q and r is less than p. Hence, uv^0xy^0z cannot be of the form ww. Hence, it is impossible to satisfy all three conditions for our choice of s, which is a contradiction to our assumption that C is context-free.

Now, by contradiction we can show that our language is also not context free.

Question 2. (12 points) Assume for contradiction that L is a context-free language. We apply the pumping lemma. Let m be the parameter of the pumping lemma. We choose to pump the string $a^m b^m c^{m^2} \in L$. We have that $a^m b^m c^{m^2} = uvxyz$, with $|vxy| \leq m$ and $|vy| \geq 1$.

We examine all the possible cases for the position of string vxy. First we note that the string v cannot span simultaneously both a^m and b^m, since if we pump up v (repeat v), the resulting string is not in the language (a's are mixed with b's). Similarly, v cannot span both b^m and c^{m2}. Therefore, it must be that v is either within a^m or b^m or c^{m2}. The same holds for y. Below are the rest of the cases. Notice that in all cases we obtain a contradiction, and therefore the language L is not context-free. The most important case is (i)

(i) v is within b^m and y is within c^{m^2}.

We have that $v = b^k$ and $y = c^l$, with $1 \leq k + l \leq m$ (since $|vxy| \leq m$ and $|vy| \geq 1$).

Consider the case where $k \geq 1$. It must be that $l < m$ (since $k + l \leq m$). From the pumping lemma we have that $uv^0xy^0z \in L$. Therefore, $a^m b^{m-k} c^{m^2-l} \in L$, and thus, it must be that $m.(m - k) = m^2 - l$.

However, this is impossible since:

$$m.(m-k) = m^2 - mk$$
$$\leq m^2 - m(\ since\ k \geq 1)$$
$$< m^2 - l(\ since\ l < m)$$

Consider now the case where $k = 0$. It must be that $l \geq 1$ (since $k + l \geq 1$). From the pumping lemma we have that $uv^0xy^0z \in L$. Therefore, $a^m b^m c^{m^2-l} \in L$, which is impossible since$m.m \neq m^2 - l$

(ii) v and y are within b^m.

If we pump down v and y we obtain a string of the form $a^m b^{m-k} c^{m^2}$, with $k \geq 1$, which obviously is not in the language.

(iii) v and y are within c^{m^2}.

If we pump down v and y we obtain a string of the form $a^m b^m c^{m2-k}$, with $k \geq 1$, which obviously is not in the language.

(iv) v and y are somewhere within $a^m b^m$.

Similar to cases (ii) and (iii).

Question 3. (10 points) **a)** $G = (V, \Sigma, R, S)$ with set of variables $V = S, W, X, Y, Z$, where S is the start variable; set of terminals $\Sigma = a, b, c$; and rules

$$S \rightarrow XY|W$$
$$X \rightarrow aXb|\epsilon$$
$$Y \rightarrow cY|\epsilon$$
$$W \rightarrow aBWBc|C|A|B$$
$$A \rightarrow aA|a$$
$$B \rightarrow cC|c$$
$$B \rightarrow bB|\epsilon$$

Here, purpose of XY is to generate the language $a^i b^i c^j$ and W generates $a^i b^j c^k$, $i \neq k$, $i, j, k \geq 0$. Purpose of X, Y is self explanatory, but in W, at first we use the first production rule in order to generate $\#a = \#b$, the count being equal to the minimum of the count of a and b and then we appropriately use A, B, C to complete the derivation.

b) First introduce new start variable S_0 and the new rule $S_0 \rightarrow S$, which gives

$$S_0 \rightarrow S$$
$$S \rightarrow BSB|B|\epsilon$$
$$B \rightarrow 00|\epsilon$$

Then we remove ϵ rules: Removing $B \rightarrow \epsilon$ yields

$$S_0 \rightarrow S$$
$$S \rightarrow BSB|BS|SB|S|B|\epsilon$$
$$B \rightarrow 00$$

Removing $S \rightarrow \epsilon$ yields

$$S0 \to S|\epsilon$$
$$S \to BSB|BS|SB|S|B|BB$$
$$B \to 00$$

We don't need to remove the ϵ-rule $S_0 \to \epsilon$ since S_0 is the start variable and that is allowed in Chomsky normal form.

Then we remove unit rules: Removing $S \to S$ yields

$$S_0 \to S|\epsilon$$
$$S \to BSB|BS|SB|B|BB$$
$$B \to 00$$

Removing $S \to B$ yields

$$S_0 \to S|\epsilon$$
$$S \to BSB|BS|SB|00|BB$$
$$B \to 00$$

Removing $S_0 \to S$ gives

$$S0 \to BSB|BS|SB|00|BB|\epsilon$$
$$S \to BSB|BS|SB|00|BB$$
$$B \to 00$$

Then we replaced ill-placed terminals 0 by variable U with new rule $U \to 0$, which gives

$$S0 \to BSB|BS|SB|UU|BB|\epsilon$$
$$S \to BSB|BS|SB|UU|BB$$
$$B \to UU$$
$$U \to 0$$

Then we shorten rules with a long RHS to a sequence of RHS's with only 2 variables each. So the rule $S_0 \to BSB$ is replaced by the 2 rules $S_0 \to BA_1$ and $A_1 \to SB$. Also the rule $S \to BSB$ is replaced by the 2 rules $S \to BA_2$ and $A_2 \to SB$. Thus, our final CFG in Chomsky normal form is

$$S_0 \to BA_1|BS|SB|UU|BB|\epsilon$$
$$S \to BA_2|BS|SB|UU|BB$$
$$B \to UU$$
$$U \to 0$$
$$A_1 \to SB$$
$$A_2 \to SB$$

To be precise, the CFG in Chomsky normal form is $G = (V, \Sigma, R, S_0)$, where the set of variables is $V = \{S_0, S, B, U, A_1, A_2\}$, the start variable is S_0, the set of terminals is $\Sigma = \{0\}$, and the rules R are given above.