2202 COL 352 Quiz2

CHINMAY MITTAL

TOTAL POINTS

9 / 10

QUESTION 1

1 NPDA **3 / 4**

+ 0 pts Incorrect/Not Attempted

✓ **+ 3 pts** *Correct NPDA construction for L1 ∩ L2*

+ 2 pts Partially correct NPDA

+ 1 pts Correctness proof

QUESTION 2

2 notCFL **3 / 3**

+ 0 pts Incorrect

+ 1 pts Correct Idea

+ 1.5 pts One case not specified

+ 2 pts Partially correct

+ 2.5 pts Unclear Explaination

✓ **+ 3 pts** *Correct*

QUESTION 3

3 Prefix **3 / 3**

✓ **+ 3 pts** *fully correct*

+ 1.5 pts correct approach but incorrect grammar

+ 0 pts completely incorrect/ unattempted

Name: CHINMAY MITTAL　　　　Roll No: 2020CS10336

(COL 352) Introduction to Automata and Theory of Computation　　　　Mar 17, 2023

Quiz 2

Duration: 40 minutes　　　　**(10 points)**

Beware: Be clear in your writing. If you use a statement proved in class or in the problem set, then write down the entire statement before using it. You will not get a new sheet, so make sure you are certain when you write something (maybe use a dark pencil). Make a judicious decision of which tool(s) to use to get a clean and short answer that fits in the space. If you cheat, you will surely get an F in this course.

---

1. **(4 points)** Show that if L_1 is a context-free language and L_2 is a regular language, then $L_1 \cap L_2$ is a context-free language by constructing an NPDA for $L_1 \cap L_2$ using the NPDA for L_1 and DFA for L_2.

There exists a DFA A for L2 $= (Q_2, \delta_2, q_0^2, F_2, \Sigma)$

set of states; transition function; start state; accepting states; input symbols

There exist a NPDA P for $L_1 = (Q_1, \delta_1, q_0^1, F_1, \Sigma, \Gamma, \perp)$

acceptance by Final state.

stack symbols; stack bottom.

We use subset construction to create and NPDA P' for $L_1 \cap L_2$

The set of states will be $Q' = Q_1 \times Q_2$

q_0 = Start state $= \{q_0^1, q_0^2\}$

F = Final states $= \{\{a, b\} \mid a \in F_1 \text{ and } b \in F_2\}$

$\Sigma' = \Sigma$

$\Gamma' = \Gamma$

$\perp' = \perp$

The transition function is as follows

if $\delta_1(q, a, A) = (q', B)$

and $\delta_2(p, a) = (p')$

Then $\delta'(\{q, p\}, a, A) = (\{q', p'\}, B)$

$\forall\, q \in Q_1,\ p \in Q_2,\ a \in \Sigma,\ A \in \Gamma$. This is a correct machine because it simulates running both the DFA and the NPDA and accepts the string only if it is accepted by both.

2020CS10336 CHINMAY MITTAL.

2. **(3 points)** Show that $L = \{a^n b^m c^n d^m \mid n, m \geq 0\}$ is not a CFL.

Proof by the contrapositive of pumping lemma for CFLs.

For any $n \geq 1$ consider the string $\omega = a^n b^n c^n d^n$, $|\omega| = 4n > n$ and $\omega \in L$.

Consider any split of ω s.t $\omega = uvwxy$ s.t $|vwx| \leq n$ and $|vx| > 0$, we show that $uv^2wx^2y \notin L$ i.e for $i = 2$ we pump ω out of L by creating uv^iwx^iy.

The string vwx cannot contain both a and c and neither can it contain both b and d. (a,c and b,d are seperated by n characters)

if vwx lies within a block ie only contains a or b or c or d then after pumping one of the following conditions will become false. $\#a = \#c$ or $\#b = \#d$ and $uv^2wx^2y \notin L$. otherwise vwx might lie in two blocks. i.e $a^n b^m c^n d^n$ again after pumping the number of a's and c's and b's and d's both cannot match hence L is not a CFL by pumping Lemma.

3. **(3 points)** Give a CFG for the following language.

$$P = \{w \mid \text{ in every prefix of w the number of a's is at least the number of b's}\}$$

$S \longrightarrow \epsilon \mid aS \mid aSbS$

production rules

can produce a string with any number of as

for every b produced and a was produced earlier

$T = \{a, b\}$ terminals

variables $V = \{S\}$

$S = S$ start symbol.

$P = \{ S \rightarrow \epsilon,$
$S \rightarrow aS,$
$S \rightarrow aSbS \}$