Csc244 Homework 3

Richard Harris

October 2021

Problem 1

Construct a NPDA recognizing G_2 from page 103 in Sipser

```
Init:
    Push($)
    Push(<S>)
    Goto Main

Main:
    If Pop(<S>):
        Goto S
    If Pop(<NP>):
        Goto NP
    If Pop(<VP>):
        Goto VP
    If Pop(<PP>):
        Goto PP
    If Pop(<CN>):
        Goto CN
    If Pop(<CV>):
        Goto CV
    If Pop(<A>):
        Goto A
    If Pop(<N>):
        Goto N
    If Pop(<V>):
        Goto V
    If Pop(<P>):
        Goto P
    If Read(a) and Pop(a):
        Goto Main
    If Read(the) and Pop(the):
        Goto Main
    If Read(boy) and Pop(boy):
        Goto Main
    If Read(girl) and Pop(girl):
        Goto Main
    If Read(flower) and Pop(flower):
        Goto Main
    If Read(touches) and Pop(touches):
        Goto Main
    If Read(likes) and Pop(likes):
        Goto Main
    If Read(sees) and Pop(sees):
        Goto Main
    If Read(with) and Pop(with):
        Goto Main
```

```
    If EoF and Pop($):
        Accept

S:
    Push(<VP>)
    Push(<NP>)
    Goto Main

NP:
    Nondeterminism:
        Push(<CN>)
        Goto Main
    Nondeterminism:
        Push(<PP>)
        Push(<CN>)
        Goto Main

VP:
    Nondeterminism:
        Push(<CV>)
        Goto Main
    Nondeterminism:
        Push(<PP>)
        Push(<CV>)
        Goto Main

PP:
    Push(<CN>)
    Push(<P>)
    Goto Main

CN:
    Push(<N>)
    Push(<A>)
    Goto Main

CV:
    Nondeterminism:
        Push(<V>)
        Goto Main
    Nondeterminism:
        Push(<NP>)
        Push(<V>)
        Goto Main

A:
    Nondeterminism:
        Push(a)
        Goto Main
    Nondeterminism:
        Push(the)
        Goto Main

N:
    Nondeterminism:
        Push(boy)
        Goto Main
    Nondeterminism:
        Push(girl)
        Goto Main
    Nondeterminism:
        Push(flower)
        Goto Main
```

```
V:

    Nondeterminism:

        Push(touches)

        Goto Main

    Nondeterminism:

        Push(sees)

        Goto Main

    Nondeterminism:

        Push(likes)

        Goto Main


P:

    Push(with)

    Goto Main
```

Problem 2

Show that the string "the girl touches the boy with the flower" has two different leftmost derivations in grammar G_2 on page 103. Describe in English the two different meanings of this sentence.

We'll be using the same shorthand from the problem above. One derivation is:

$< S > \to < NP >< VP > \to < CN >< VP >$

$\to < A >< N >< VP > \to$ the $< N >< VP > \to$ the girl $< VP >$

$\to$ the girl $< CV >< PP > \to$ the girl $< V >< NP >< PP > \to$ the girl touches $< NP >< PP >$

$\to$ the girl touches $< CN >< PP > \to$ the girl touches $< A >< N >< PP >$

$\to$ the girl touches the $< N >< PP > \to$ the girl touches the boy $< PP >$

$\to$ the girl touches the boy $< P >< CN > \to$ the girl touches the boy with $< CN >$

$\to$ the girl touches the boy with $< A >< N > \to$ the girl touches the boy with the $< N >$

$\to$ the girl touches the boy with the flower

Another derivation is:

$< S > \to < NP >< VP > \to < CN >< VP >$

$\to < A >< N >< VP > \to$ the $< N >< VP > \to$ the girl $< VP >$

$\to$ the girl $< CV > \to$ the girl $< V >< NP > \to$ the girl touches $< NP >$

$\to$ the girl touches $< CN >< PP > \to$ the girl touches $< A >< N >< PP >$

$\to$ the girl touches the $< N >< PP > \to$ the girl touches the boy $< PP >$

$\to$ the girl touches the boy $< P >< CN > \to$ the girl touches the boy with $< CN >$

$\to$ the girl touches the boy with $< A >< N > \to$ the girl touches the boy with the $< N >$

$\to$ the girl touches the boy with the flower

The two different meanings are "the girl touches the boy using a flower" and "the girl touches the boy who has a flower." Hopefully this is right, I pursue math and computer science for a reason.

Problem 3

Show that neither DCFL nor CFL are closed under intersection

Proof. We know the language $C = \{a^n b^n c^n \mid n \geq 0\}$ is not context free. Consider the two languages $A = \{a^m b^n c^n \mid n, m \geq 0\}$ and $B = \{a^n b^n c^m \mid n, m \geq 0\}$ whose intersection is C. Since DCFLs are a subset of CFLs, then it would suffice to create a DPDA for each A and B to show that neither DCFL nor CFL are closed under intersection. First comes A, then B:

```
Init:
    Push($)
    Goto Count_a

Count_a:
    If EoF and Pop($):
        Accept
    If Read(a):
        Goto Count_a:
    If Read(b):
        Push(b)
        Goto Count_b:
    If Read(c):
        Reject

Count_b:
    If EoF:
        Reject
    If Read(a):
        Reject
    If Read(b):
        Push(b)
        Goto Count_b
    If Read(c) and Pop(b):
        Goto Count_c

Count_c:
    If EoF and Pop($):
        Accept
    If Eof and Pop(b):
        Reject
    If Read(c) and Pop($):
        Reject
    If Read(a):
        Reject
    If Read(b):
        Reject
    If Read(c) and Pop(b):
        Goto Count_c
```

```
Init:
    Push($)
    Goto Count_a

Count_a:
    If EoF and Pop($):
        Accept
    If Read(a):
        Push(a)
        Goto Count_a:
    If Read(b) and Pop(a):
        Goto Count_b:
    If Read(c) and Pop(a):
        Reject
    If Read(c) and Pop($):
        Goto Count_c

Count_b:
    If EoF and Pop($):
        Accept
    If Read(a):
        Reject
    If Read(b) and Pop(a):
        Goto Count_b
    If Read(c) and Pop($):
        Goto Count_c
    If Read(c) and Pop(a):
        Reject

Count_c:
    If EoF:
        Accept
    If Read(a):
        Reject
    If Read(b):
        Reject
    If Read(c):
        Goto Count_c
```

Thus, A and B are DCFLs and CFLs. Yet their intersection is not context free. Thus, DCFL and CFL are not closed under intersection. □

Problem 4

Use Problem 1 and DeMorgan's Law (Theorem 0.20) to show that the class of context free languages is not closed under complementation.

Proof. Suppose by contradiction that CFL is closed under complementation. Then, using A and B from the problem above, we have $A \in \text{CFL}$ and $B \in \text{CFL}$. Likewise, by our assumption, $A^c \in \text{CFL}$ and $B^c \in \text{CFL}$. From class we know that CFL is closed under union, therefore it is true that $A^c \cup B^c \in \text{CFL}$. Taking the complement of this and using DeMorgan's Law we have $(A^c \cup B^c)^c = A \cap B$. But we already proved that $A \cap B$ is not in CFL. By contradiction, CFL is not closed under complementation. □

Problem 5

Show that DCFL is not closed under union.

Proof. From class, it is given that DCFL is closed under complementation and above we have proven that it is not closed under intersection. Suppose by contradiction that DCFL is closed under union. Then, using A and B from Problem 3, we have $A, A^c, B, B^c \in$ DCFL. Using our assumption and DeMorgan's Law, we have $(A^c \cup B^c)^c = A \cap B$. But $A \cap B$ is not in DCFL. By contradiction, DCFL is not closed under union. □

Problem 6

Let $A/B = \{w \mid wx \in A$ for $x \in B\}$. Show that if A is context free and B is regular, then A/B is context free.

Proof Idea: We are given that N_1 is a NPDA recognizing A and that D is a DFA recognizing B. Using these, we want to construct an NPDA N_2 recognizing A/B. Since B is regular, there are a finite amount of $x \in B$. Reconstruct N_1 so that it ends by running D. To construct N_2, take N_1 and when a transition goes into D, make it go straight to accept instead.

Proof. Let $N_1 = (Q_1, \Sigma, \Gamma, \delta_1, q_1, F_1)$ be a NPDA recognizing A, $D = (Q_2, \Sigma, \delta_2, q_2, F_2)$ be a DFA recognizing B. Given the relation between A and B, we construct N_2 by taking N_1 and making a new set of accept states as defined by $\{r \mid r$ is a state in N_1 where if you start from that state and process a string from D, you reach an accept state$\}$.

□

Problem 7

We defined the CUT of language A to be $CUT(A) = \{yxz \mid xyz \in A\}$. Show that the class of CFLs is not closed under CUT.

Proof. Consider the language $L = \{a^i b^j c b^j a^i \mid i, j \geq 0\}$. We know L is context free because we have already proven that $\{w\#w^R \mid w \in \{a,b\}^*\}$ is context free and this is the special case where $w = a^i b^j$ and $\# = c$. By contradiction, suppose CFLs are closed under the CUT operation, then $CUT(L)$ must have a pumping length p. A valid string in $CUT(L)$ is $b^p a^p b^p c a^p$ from $a^p b^p c b^p a^p \in A$ using $x = a^p b^p c$, $y = b^p$, $z = a^p$. Let's enumerate all the possibilities for which the string can be broken into $s = uvxyz$ as per the Pumping Lemma:

1.) uxy is contained in the first b^p,

2.) uxy is between the first b^p and the first a^p,

3.) uxy is contained in the first a^p,

4.) uxy is between the first a^p and the second b^p,

5.) uxy is contained in the second b^p,

6.) uxy is between the second b^p and the second a^p and $c \in x$ (cannot pump the c),

7.) uxy is contained in the second a^p.

Noticing how there is no way to keep the two sets of a's and b's even, pumping this string would throw it out of $CUT(A)$. Thus, $CUT(A)$ is not a CFL and CFLs are not closed under CUT. □

Problem 8

Show that CFL is closed under (a) concatenation and (b) Kleene star.

a.) *Proof.* Let $L(L)$,$L(M)$ be CFLs defined by the CFGs L and M. Then, L and M have a start variable $S_1 \in L$ and $S_2 \in M$. For the language $L(L)L(M)$, we take all the rules from L and M and put them together while adding only one rule: $S \to S_1 S_2$. Thus, we can construct a CFG for any concatenation of two languages. □

b.) *Proof.* Let $L(L)$ be a CFL defined by the CFG L. Then, L must have a start variable $S \in L$. To construct $(L(L))^*$, add the rule $S \to SS$. Thus, we can construct a CFG for the Kleene star of a language. □

Problem 9

Show $\{a^n b^n c^n \mid n \geq 0\}$ is context sensitive.

Proof. Construct the grammar below:

$$S \longrightarrow aAbc|\varepsilon$$
$$aAb \longrightarrow aaAbXbc$$
$$cXb \longrightarrow bc$$
$$A \longrightarrow \varepsilon$$
$$bXb \longrightarrow bb$$

□

Now let us demonstrate that this grammar generates the language above using the string $a^3b^3c^3 = aaabbbccc$. We should note that this grammar was designed for left derivation only.

$$S \longrightarrow aAbc \longrightarrow aaAbXbcc$$
$$\longrightarrow aaaAbXbcXbcc \longrightarrow aaabXbcXbcc$$
$$\longrightarrow aaabbcXbcc \longrightarrow aaabbbccc = a^3b^3c^3$$

Extra Credit

Consider a Same Game type of game in one dimension with 2 colors. Let $L = \{w \mid w \in \{r, b\}^*$ and w is winnable$\}$. Show that L is context free.

To start, we've won the game once we get to the string ε. The step before this string must have been either r^x or b^x for $x \geq 2$. WLOG, we'll focus on r^x. Then the step before would have to be $r^i b^y r^j$ such that $i + j = x$ for i,$j \geq 0$ and $y \geq 2$. Suppose we had rules

$$R \longrightarrow r^x \mid r^i B r^j$$
$$B \longrightarrow b^y \mid b^i R b^j$$

where $i + j = x$ or $i + j = y$. These two rules will only define winning states. Then the start rule would be $S \longrightarrow R \mid B \mid \varepsilon$. Next on our list would be to properly define R and B.