Computational Models – Exercise 7

Due Saturday, 03 June 2023

Each student must solve the problems on their own. If you encounter difficulties, you may ask a classmate for a hint or the general idea. However, detailed discussion, note-taking, or sharing of written solutions is not allowed. Do not write down your answers while communicating with other people or show the answers for feedback.

Our grading app has severe limitations, such as no zoom tool. To make sure we can grade your work, please follow these technical guidelines:

Submit a **single PDF file** through Moodle.
The file size is limited to **10 MB**. If necessary, google *reduce PDF file size*.
Fill in your answers **on this form*** in the allocated spaces. The space provided gives you an indication of the expected length and level of detail of the answer. You may add a little more space if you need.
Include everything from this form in your submission. In particular, **include the problem statements**. Do not delete any text or omit pages, just add your answers.
Ensure your answers are **legible** (easy to read) at zoom 100% on a standard computer screen. Your text should be **large**, **sharp**, and in **high contrast** with the background.
Do not squeeze scanned solutions to fit in the space, as the text will become small.
Verify that pages are properly **ordered** and **oriented**.
The page size must be **A4**. Before submitting your file, check its page size using Acrobat Reader: go to File > Properties > Description and confirm that Page Size is around 21 × 29 cm. Note that scanning A4 pages does not guarantee the resulting page size will be A4, due to scaling. If necessary, google *resize PDF to A4*.
Do not add your answers as PDF comments. If you can drag them in Acrobat Reader, they are comments. If necessary, google *flatten PDF*.

A **5-point bonus** will be given to solutions typed in a word processor. Hand-sketched illustrations or diagrams will not deny you this bonus.
If there are technical issues with your submission, you may receive a fine. In extreme cases, your submission may not be graded at all.
If you need help or have questions, please use the course forum at Piazza.

*The only exception is in case you use LaTeX or a similar typesetting system. In that case, copy-paste everything from this file, except for illustrations or other hard-to-reproduce graphical elements. No need to fix corrupted formulas.

Worked with Jemma Diamond - 806839

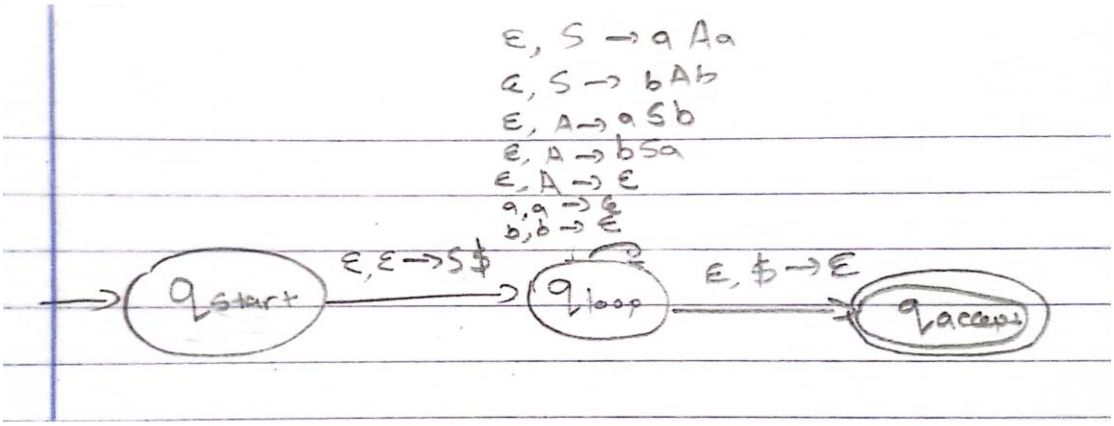
Problem 1 (10 pts)

Build a PDA for each of the following grammars using the procedure taught in class.

a.

$S \rightarrow aAa \mid \ bAb$

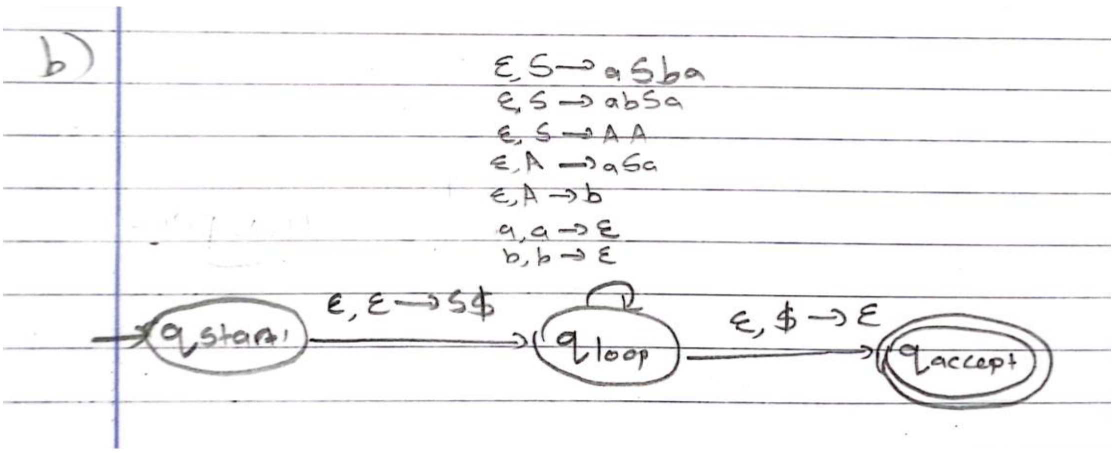
$A \rightarrow aSb \mid bSa \mid \varepsilon$



b.

$S \rightarrow aSba \mid abSa \mid AA$

$A \rightarrow aSa \mid b$



Problem 2 (35 pts)

Prove whether the following languages are CFL or not. If they are CFL prove by presenting a CFG/PDA or NFA/RegExp. If they are not CFL prove using the pumping lemma for CFL.

a. $L = \{a^n b^m c^n d^m \mid n, m \geq 0\}$

ATC that L is a CFG, therefore the pumping lemma holds for L. Let p be the pumping length for L. Let w=$a^p b^p c^p d^p$. W ∊ L and |w| > p .Divide w into five parts uvxyz s.t |vy| > 0 and |vxy| <= p. (by the pumping lemma such a division exists). Since |vxy| <= p, this results in only 2 cases:

1. v and y are homogenous:

v and y contain only one type of alphabet symbol. By choosing i=2 we get $w_2 = uv^2xy^2z$ in which the number of appearances of one or two symbols increased (|vy|>0) while the number of appeareances of the other 3 or 2 symbols remain unchanged. So $w_2$cannot contain the same number of a's, b's, c's and d's. Therefore $w_2 \notin$L.

2. v or y (not both) are heterogenous:

v or y (but not both) contain two types of alphabet symbols. (not both because |vxy| <= p). So if we choose i=2 then the order of the symbols in v^2 or y^2 is destroyed (i.e no longer one symbol and then the next) and $w_2 = uv^2xy^2z \notin$L.

Therefore in both cases there is a contradiction and therefore L is not a CFG.

b. $L = \{a^m b^n c^k \mid m,n,k \geq 0,\ k \leq \min(m,n)\ \}$.

ATC that L is a CFG, therefore the pumping lemma holds for L. Let p be the pumping length for L. Let w=$a^p b^p c^p$. W ∊ L and |w| > p .Divide w into five parts uvxyz s.t |vy| > 0 and |vxy| <= p. (by the pumping lemma such a division exists). Since |vxy| <= p, this results in only 2 cases:

1. v and y are homogenous:

v and y contain only one type of alphabet symbol. By choosing i=2 we get $w_2 = uv^2xy^2z$ in which the number of appearances of one or two symbols increased (|vy|>0) while the number of appearances of the other 1 or 2 symbols remain unchanged. If c is one of the symbols which gained an appearance, then it has one more appearance than either a or b (dependent on which other symbol gained an appearance). Therefore c is no longer the symbol with the least number of appearances and therefore $w_2 \notin$L.

2. v or y (not both) are heterogenous:

v or y (but not both) contain two types of alphabet symbols. (not both because |vxy| <= p). So if we choose i=2 then the order of the symbols in v^2 or y^2 is destroyed (i.e no longer one symbol and then the next) and $w_2 = uv^2xy^2z \notin$L.

Therefore in both cases there is a contradiction and therefore L is not a CFG.

c. $L = \{wvw \mid w, v \in \{a,b\}^*, |w| = |v| \}$

ATC that L is a CFG, therefore the pumping lemma holds for L. Let p be the pumping length for L. Let w=$a^p b^p a^p$. W ∊ L and |w| > p .Divide w into five parts uvxyz s.t |vy| > 0 and |vxy| <= p. (by the pumping lemma such a division exists).

|vy| must be a multiple of 3 otherwise $w_i = uv^i xy^i z$ cannot be divided into 3 substrings of equal length for each i.

Since |vxy| <= p, there are only 2 cases:

1. vy is homogenous (i.e vy = a^{3s}or vy = b^{3s}):

If vy is comprised of a's from the first 'a' , then let w2 = $uv^2xy^2z = a^{3s+p}b^pa^p$. The size of w2 is 3(p+s) and therefore w= a^{p+s}, v =$a^{2s}b^{p-s}$ and the last w = b^sa^p. Therefore the first w != the last w and therefore w2 is not part of L.

If vy is comprised of b's , then let w2 = $a^pb^{p+3s}a^p$. The size of w2 is 3(p+s) and therefore w= a^pb^s, v = b^{p+s} and the last w = b^sa^p. Therefore the first w != the last w and therefore w2 is not part of L.

If vy is comprised of a's from the last 'a', then let w2 = $a^pb^pa^{p+3s}$. The size of w2 is 3(p+s) and therefore w= a^pb^s, v =$b^{p-s}a^{2s}$ and the last w =a^{p+s}. Therefore the first w != the last w and therefore w2 is not part of L.

2. vy is heterogenous:

If vy is a^sb^t (0 < s+t <= p), let s+t = 3r. Let w2= $a^{p+s}b^{p+t}a^p$. The size of w2 is 3p + s + t = 3(p+r) and therefore w= a^pq1…qr (q represent some symbol) , v =q1…q2r b^{p-r} and therefore the last w = b^ra^p. Therefore the first w != the last w and therefore w2 is not part of L.

If vy is b^sa^t (0 < s+t <= p), let s+t = 3r. Let w2= $a^pb^{p+s}a^{p+t}$. The size of w2 is 3p + s + t = 3(p+r) and therefore w= a^pb^r , v =b^{p-r}q1…qr (q represent some symbol) and therefore the last w = q1…q2r a^p. Therefore the first w != the last w and therefore w2 is not part of L.

Therefore in both cases there is a contradiction and therefore L is not a CFG.

d. $L = \{a^m b^n \mid n = m^2, \ \ n, m \geq 0\} \ \}$

I don't know

e. $L = \left\{a^i b^j c^k \ \middle| \ not(i = j = k)\right\}$

S -> A | B

A -> CD

B -> EQ

C -> aCb | aE |bF

D -> cD | epsilon

E -> aE | epsilon

F -> bF | epsilon

Q -> bQc | bF | cR

R -> cR | epsilon

Problem 3 (40 pts)

Prove or disprove the following claims:

a. If L is a CFL then $\{ww^R \mid w \in L\}$ is **not** CFL

False. Counterexample:

Let L= $\{a^{2n}\}$. This is a CFL as it can be written as S -> aSa | epsilon. Therefore $\{ww^R \mid w \in L\} = \{a^{4n}\}$ which is also a CFL as it can be written as S -> aaSaa | epsilon.

b. If L is **not** CFL then L^* is **not** CFL

False. Counterexample:

Let L= $\{a^p$ | p is a prime number}. We know L is not a CFL. But L*={$(a^p)^*$ | p is a prime number} is a CFL as L* consists of the empty string ε, strings with prime lengths, and their concatenations which results in a* which is a CFL.

c. If L is CFL then $L \cap L^R$ is CFL.

False. Counterexample:

Let L = $\{a^n b^n a^k$ | n,k>=0}, we know L is a CFL as it can be represented as a PDA. Let L^R = { $a^k b^n a^n$ | n,k>=0}, we know L^R is a CFL as it can be represented as a PDA. But L ∩ L^R = { $a^n b^n a^n$| n>=0} is not a CFL as 2 stacks are needed for its PDA and therefore a PDA can not be produced.

d. If L is **not** CFL and L' is finite, then $L' \cup L$ is **not** CFL.

True. ATC that L′ ∪ L is a CFL. Since L' is finite, this means L' is a RL and therefore closed under compliment – therefore L'^C is a RL. L'^C∩ L′ ∪ L = L is a CFL (CFL is closed under intersection with a RL). This is a contradiction to L not being a CFL.

e. If L_1 is CFL and L_2 is regular, then $L_1\ XOR\ L_2$ is CFL.

$(L_1\ XOR\ L_2 = \{w \mid w \in L_1\ or\ w \in L_2\ \ and\ w \notin L_1 \cap L_2\ \ )$

False. Counterexample:
Let L1= $\{a^i b^j c^k \mid not(i = j = k)\}$ is a CFL (proven in problem 1e). Let L2 ={ $a^i b^j c^k$ | i,j,k >=0} is a RL. L1 XOR L2 = { $a^n b^n a^n$| n>=0} which is not a CFL.

Problem 4 (6 pts)

Let L be a RL such that the shortest word in L is of length $2^k + 1$.

Let A be a DFA that recognizes L.

Prove that the number of states of A is greater than $2^k + 1$.

Let w be the shortest word in L. Therefore w can be computed by A. w will go through $2^k + 2$ states throughout it's computation, as for each symbol it makes a transition to a new state (which is $2^k + 1$ states) and you take into account the initial state, resulting in $2^k + 2$. Each one of those states is unique, as if one of them wasn't, then that would mean at least one of the states was revisited and there was a loop. However this means that a shorter word than w can be computed by removing the loop from the computation, which is a contradiction. Therefore the number of states in the DFA is at least $2^k + 2$.

Problem 5 (9 pts)

Let L be a context-free language such that the shortest word in L is of length $2^k + 1$. Let G be a CFG in CNF that generates L. Prove that the number of variables in G is greater than k.

Denote w as the shortest word in L. The word w can be derived in $2(2^k + 1)$ - 1 steps since G is in CNF. The number of leaves in the parsing tree is $2^k + 1$ as the number of leaves correspond to terminals (symbols) that make up the word, of which there is $2^k + 1$. Therefore the number of variables used in the parsing tree is the number of steps minus the number of leaves which is $2(2^k + 1)$ - 1 - $(2^k + 1) = 2^k$. All of those variables are unique as if they weren't, that means that there was a repetition of a variable down the parsing tree, and it is possible to remove this repetition (i.e moving the subtree rooted at the second occurrence of the repeated variable all the way up and rooting it in the first occurrence) and therefore make a shorted word. This is a contradiction to w being the shortest word. Therefore every variable in the parsing tree is unique and therefore there is $2^k > k$ variables.