CSci 435: Formal Languages and Automata

Instructor: Dr. M. E. Kim Name: \_Grant Haataja \_

**Home Assignment 6: 90 points + 20 points (optional)**

In any (N/D)PDA, assume that a start stack symbol z is already in the stack; so, you don’t have to insert z into the stack at the beginning of transition.

Q1.[30] Prove if the following languages are CFL or not.

If L is a CFL, give its CFG. Otherwise, prove it by Pumping Lemma.

If any closure property of CFL is applicable, apply them to simplify it before its proof.

1. [10] L = {*wwRw* | *w* ∈ {*a, b*}\*}

*Ans*. Suppose the above language is context free. Then the Pumping Lemma holds. Given an arbitrary positive integer *m*, we select a string *wwRw* such that | *wwRw* | ≥ *m*. Let *wwRw* = *ambmbmamambm* ∈L. We split the string into substrings such that *w* = *uvxyz*. We choose *u* = *aj*, *v* = *ak*, *x* = *am-j-kbp*, *y* = *bq*, *z* = *bm-p-qbmamambm* where *m* ≥ *j, k, p, q* > 0 and |*vxy*| = *m* – *j* + *p* + *q* = *m* – (*j* – *p* – *q*) ≤ *m*.

By Pumping Lemma, *wi* = *uvixyiz*. So, by pumping with *i* = 0, *w0* = *uv0xy0z* = *uxz* = *am-kbm-qbmamambm*. This is clearly not in L since *m* – ­­*k* ≠ *m* and *m* – ­­*q* ≠ *m*. This contradicts the Pumping Lemma, which proves that L is not context free.

1. [10] L = { *anwwRbn* | *n* ≥ 0, *w* ∈ {*a, b*}\*}

*Ans*. Suppose the above language is context free. Then the Pumping Lemma holds. Given an arbitrary positive integer *m*, we select a string *anwwRbn* such that | *anwwRbn* | ≥ *m*. Let *anwwRbn* = *amambmbmambm* ∈L. We split the string into substrings such that *w* = *uvxyz*. We choose *u* = *aj*, *v* = *ak*, *x* = *am-j-kap*, *y* = *aq*, *z* = *am-p-qbmbmambm* where *m* ≥ *j, k, p, q* > 0 and |*vxy*| = *m* – *j* + *p* + *q* = *m* – (*j* – *p* – *q*) ≤ *m*.

By Pumping Lemma, *wi* = *uvixyiz*. So, by pumping with *i* = 0, *w0* = *uv0xy0z* = *uxz* = *am-kam-qbmbmambm*. This is clearly not in L since *m* – ­­*k* ≠ *m* and *m* – ­­*q* ≠ *m*. This contradicts the Pumping Lemma, which proves that L is not context free.

1. [10] L = {*anbjajbn* | *n* ≥ 0, *j* ≥ 0}

*Ans*. Suppose the above language is context free. Then the Pumping Lemma holds. Given an arbitrary positive integer *m*, we select a string *w* such that | *w* | ≥ *m*. Let *w* = *ambmambm* ∈L. We split the string into substrings such that *w* = *uvxyz*. We choose *u* = *aj*, *v* = *ak*, *x* = *am-j-kbp*, *y* = *bq*, *z* = *bm-p-qambm* where *m* ≥ *j, k, p, q* > 0 and |*vxy*| = *m* – *j* + *p* + *q* = *m* – (*j* – *p* – *q*) ≤ *m*.

By Pumping Lemma, *wi* = *uvixyiz*. So, by pumping with *i* = 0, *w0* = *uv0xy0z* = *uxz* = *am-kbm-qambm*. This is clearly not in L since *m* – ­­*k* ≠ *m* and *m* – ­­*q* ≠ *m*. This contradicts the Pumping Lemma, which proves that L is not context free.

1. [10, optional] L = { *an*| *n* is a prime number }

*Ans.* Suppose the above language is context free. Then the Pumping Lemma holds. Given an arbitrary positive integer *m*, we select a string *w* such that | *w* | ≥ *m*. Let *w* = *an* where *n* is a prime number. Now, *ap* = *uvxyz*, so we have that *v* = *ak* and *y* = *al*, where *k* + *l* ≥ 1. By the pumping lemma, we have *uv1+pxy1+pz* ∈L, which means that *ap+kp+lp* ∈L. Hence, *ap*(1 + *k* + *l*) ∈L, which is a contradiction since *p*(1 + *k* + *l*) is not prime. Therefore, L is not context free.

Q2. [20] Prove that the following languages are linear or not.

If L is linear, give the linear-CFG for L. Otherwise, prove it by Pumping Lemma for a Linear-CFL.

1. [10] L = { *w* | *na*(*w*) + *nb*(*w*) = *nc*(*w*) } is not linear.

*Ans*. Suppose that L is linear, so the pumping lemma must hold. . Given an arbitrary positive integer *m*, we select a string *w* such that | *w* | ≥ *m*. Let *w* = *ambmc2m* ∈L. We split the string into substrings such that *w* = *uvxyz*. We choose *u* = *aj*, *v* = *ak*, *x* = *am-j-kbmc2m-p-q*, *y* = *cp*, *z* = *cq*. By Pumping Lemma, *wi* = *uvixyiz*. So, by pumping with *i* = 0, *w0* = *uv0xy0z* = *uxz* = *am-kbmc2m-p*. This is not in L if *p* ≠ *k*. This contradicts the Pumping Lemma, which proves that L is not linear.

1. [10] L = { *anbmcn* | *n, m* ≥ 0 } ∪ { *anbncm* | *n, m* ≥ 0 } is linear or not.

*Ans*. Suppose that L is linear, so the pumping lemma must hold. . Given an arbitrary positive integer *m*, we select a string *w* such that | *w* | ≥ *m*. Let *w* = *ambmcm* ∈L. We split the string into substrings such that *w* = *uvxyz*. We choose *u* = *aj*, *v* = *ak*, *x* = *am-j-kbmcm-p-q*, *y* = *cp*, *z* = *cq*. By Pumping Lemma, *wi* = *uvixyiz*. So, by pumping with *i* = 0, *w0* = *uv0xy0z* = *uxz* = *am-kbmcm-p*. This is not in L if *p* ≠ *k*. This contradicts the Pumping Lemma, which proves that L is not linear.

Q3. [30] Prove the following properties clearly.

1. [10] The family of CFLs is closed under reversal.

*Ans*. Let L be a context free language, and G be a Chomsky Normal Form grammar that satisfies L. We denote the reversal of L as LR. The grammar for LR can be obtained by reversing all productions in G, so A ⟶ BC becomes A ⟶ CB, and so on. We do not reverse the productions of the form A ⟶ *a*. This new grammar is GR, and will derive the reverse strings of all the strings in L, as expected.

1. [10] The family of DCFL is closed under regular difference:

i.e. for a DCFL L1 and a RL L2, L1 − L2 ∈ DCFL.

*Ans*. Let L1 be a Deterministic Context Free Language, and let L2 be a Regular Language. Theorem 8.5 from the slides states that if L1 is a CFL and L2 is a RL, then L­1 ⋂ L2 is context free.

Now, L1 – L2 = L1 ⋂ , and the family of regular languages is closed under complements. So, since L2 is a regular language, is also a regular language. Thus, by Theorem 8.5, L1 ⋂ is context free. Therefore, L1 – L2 is also context free, which proves that the family of DCFL is closed under regular difference.

1. [10] The family of CFLs is not closed under complement. Give an example for it.

*Ans*. Let L1 and L2 be Context Free Languages. Let and be the complements of L1 and L2, respectively. Suppose that and are context free. Then, by Theorem 8.3, ⋃ = L3, where L3 is a CFL. Now, by our previous assumption, we suppose further that is context free. By DeMorgan’s Law, we have = = L1 ⋂ L2. However, as stated in Slide 10, CFLs are not closed under intersection. Therefore, CFLs are not closed under complement.

1. [10] If L1 is linear and L2 is regular, L1⋅L2 is a linear language.

*Ans*. Let L1 be a linear language and L­2 be a regular language. We know that the family of regular languages is closed under concatenation. We also know that a grammar for a regular language is a grammar with productions of the form A → *x*B or A → B*x* or A → *x*. Furthermore, a linear language has grammar with productions of the form A → *x*B*y*.

In any case of the regular grammar, concatenation with the linear grammar will produce productions of the form *a*V*b* where V is a variable and *a* and *b* are terminals. This shows that L1⋅L2 is a linear language.

1. [10, optional] The family of DCFLs is **not** closed under reversal. Give an example.

*Ans*.

Example: L = {*aibjck* | *i* < *j*} ⋃ {*aibjck* | *i* < *k*} is a DCFL with 0 and 1 as a stack language to ensure that *i* < *j* and *i* < *k*, respectively. However, for LR, the 0 and 1 won’t be able to guide the order properly, so LR is not a DCFL.