Author: Ragesh Jaiswal

1.1 Closure Properties of CFL

In this section we take up some important closure properties related to CFLs.

Claim 1.1.1 *The class of CFLs is closed under the union ($\cup$) operation.*

Proof Idea: We need to pick up *any* two CFLs, say L_1 and L_2 and then show that the union of these languages, $L_1 \cup L_2$ is a CFL. But how do we show that a language is a context free? One method for this is to come up with a CFG and show that the grammar generates the language. We know that since L_1 and L_2 are CFLs, we have some CFGs say G_1 and G_2 that generate L_1 and L_2 respectively. So we may try to use these two grammars to construct the grammar that generates $L_1 \cup L_2$. In the exam you need not write the proof idea. This is just for conveying the idea. You need to write the proof as below.

Proof: Let L_1, L_2 be any two CFL, we will show that $L = L_1 \cup L_2$ is a CFL. Since L_1, L_2 are CFLs, there must exist CFGs which generate these two languages. Let G_1 and G_2 generate the languages L_1 and L_2 respectively, where:

$$G_1 = (V_1, \Sigma_1, R_1, S_1), \text{ and}$$

$$G_2 = (V_2, \Sigma_2, R_2, S_2)$$

We assume that the sets V_1 and V_2 are disjoint, or $V_1 \cap V_2 = \phi$ (we can always assume this because if the sets are not disjoint we can make them so, by renaming variables in one of the grammars). Consider the following grammar:

$$G = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, R_1 \cup R_2 \cup \{S \to S_1 | S_2\}, S)$$

The above grammar is basically a combination of the grammars G_1 and G_2 in which we have added the new start state S and a new production rule $S \to S_1|S_2$. Now we need to show that G generates L(this is basically the proof of correctness of the construction). For this we need to show the following two things:

1. For any string $s \in L$, G generates s: We know that either $s \in L_1$ or $s \in L_2$ which implies that either $S_1 \Rightarrow^* s$ or $S_2 \Rightarrow^* s$. Since G has the production $S \to S_1|S_2$ we can conclude that $S \Rightarrow^* s$. So G generates s.

2. Let s be any string generated by G, then $s \in L$: We have $S \Rightarrow^* s$, this means that either $S_1 \Rightarrow^* s$ or $S_2 \Rightarrow^* s$. Now since we have made sure that $V_1 \cap V_2 = \phi$, s is either derived from S_1 using the rules R_1 only or it is derived from S_2 using rules R_2 only. This means that $s \in L_1 \cup L_2$.

■

Claim 1.1.2 *The class of CFLs is closed under the star operation (*).*

Proof Idea: Pick up any CFL L_1, we need to show $L = L_1^*$ is a CFL. Let $G_1 = (V_1, \Sigma_1, R_1, S_1)$ be the grammar that generates L_1. Consider the new grammar $G = (V_1 \cup \{S\}, \Sigma_1, R_1 \cup \{S \to SS|S_1|\epsilon\}, S)$, now show that this grammar generates L.

Claim 1.1.3 *The class of CFLs is* not *closed under intersection.*

Proof Idea: We use pumping lemma to prove this. Pumping lemma would be covered in the next section.

Problems to think about: Prove the following:

Claim 1.1.4 *The class of CFLs is* not *closed under complement operation.*

Hint: You can show this in 3 lines using de morgan's law and claims 1.1.1 and 1.1.3.

Prove the following:

Claim 1.1.5 *The class of CFLs is* not *closed under the difference operation. (For any two sets A, B the difference operation is simply $A - B$, or the set of elements that are in A but not in B).*

Hint: Remember for disproving things you just need to come up with an example. Claim 1.1.4 might be helpful in this.

1.2 The Pumping Lemma

Let us restate the pumping lemma for our reference:

If A is a context-free languages, then there is a number p (pumping length) where, if s is any string in A of length at least p, then s may be divided into 5 pieces $s = uvxyz$ satisfying the conditions:

1. For each $i \geq 0$, $uv^ixy^iz \in A$
2. $|vy| > 0$, and
3. $|vxy| \leq p$.

We will not go over how we derive the pumping lemma but instead we will concentrate on how to apply it to show that some language is *not* CFL. The best way to do is to pick up a problem.

Problem 2.18a: Show that the language $A = \{0^n1^n0^n1^n : n \geq 0\}$ is not context free.

Proof Sketch: We start by assuming that A is context free. This means that the pumping lemma should hold for A. Now we come up with a string $s \in A$ of length at least p such that the condition 1 is not satisfied for any breakup of s satisfying conditions 2 and 3. This results in the contradiction which shows that our initial assumption was incorrect or A is not context free. Since we do not know the pumping length, assume some pumping length p (in effect we will end up showing that our proof holds for *any* value of p). Now choose s which has a size at least p. Let $s = 0^p1^p0^p1^p$, now to simulate the effect of breaking up the string in all possible ways such that conditions 2 and 3 are satisfied we can hypothetically move a *window* of length p (the window represent the string vxy) over s and try pumping and checking if condition 1 is satisfied. It so happens that that condition

1 is not satisfieds for any breakup as we will see in the proof. Let me emphasize once again that the proof idea is not to be written in the exam, this is to convey the idea. You just need to write the proof as below.

Proof: For the sake of contradiction let us assume that A is context free. This means that for any string $s \in A$ of length at least the pumping length (say p), s can be written as $uvxyz$ and the 3 conditions of the lemma is satisfied. Let $s = 0^p1^p0^p1^p$, now we do the following case analysis for the breakup of $s = uvxyz$, with $|vy| > 0$ and $|vxy| \leq p$:

1. vxy contains either only 0s or only 1s: in this case $uv^2xy^2z \notin A$ since this has unequal number of 0s and 1s.
2. vxy spans two symbols: This means that vxy spans the initial 0^p1^p or 1^p0^p or the trailing 0^p1^p. In any of these cases $uv^2xy^2z \notin A$ since either the order of 0s and 1s are disturbed or the number of 0s and 1s are not of the form required.

We thus get a contradiction which shows that A is not context free. ∎

Problem 2.18c: Show that the language $A = \{\omega\#x : \omega \text{ is a substring of } x, \text{ where } \omega, x \in \{a, b\}^*\}$

Hint: Consider the string $s = a^pb^p\#a^pb^p$, where p is the pumping length.

Something to think about: Can we use the string $s = a^p\#a^p$ instead? Why or why not?

Claim 1.2.1 *The class of CFLs is* not *closed under the intersection ($\cap$) operation.*

Proof Idea: Remember that to show the class is not closed under $\cap$, we just need to come up with two languages that are context free but their intersection is not context free. Consider the following two languages:

$$A = \{a^mb^nc^n : m, n \geq 0\}$$

$$B = \{a^mb^mc^n : m, n \geq 0\}$$

We find that $A \cap B = \{a^nb^nc^n : n \geq 0\}$ which can be shown to be not context free by using the pumping lemma.