Theory of Automata – Homework 7

Name – Akshay Kumar Singh

R11603620

1. **Use the pumping theorem to show that the following languages are not context-free**

   a). $\{www: w \in \{a, b\}^*\}$
   b). $\{w \in \{a, b, c\}^*: w$ **has equal number of** $a's, b's \ and \ c's\}$

Sol : **(a)** L = {www : w ∈ {a, b}*}. The easiest way to do this is not to prove directly that L = {www : w ∈ {a, b}*} is not context free.
Instead, let's consider L1 = L ∩ a*ba*ba*b. If L is context free, L1 must also be. L1 = {an ban ban b : n ≥ 0}. To show that L1 is not context free, let's choose w = a^m ba^m ba^m b. First we observe that neither v nor y can contain b, because if either did, then, when we pump, we'd have more than three b's, which is not allowed.
So both must be in one of the three a regions. We consider the cases:
(1, 1) That group of a's will no longer match the other two, so the string is not in L1.
(2, 2) That group of a's will no longer match the other two, so the string is not in L1
(3, 3) That group of a's will no longer match the other two, so the string is not in L1
(1, 2) At least one of these two groups will have something pumped into it and will no longer match the one that is left out.
(2, 3) At least one of these two groups will have something pumped into it and will no longer match the one that is left out.
(1, 3) excluded since |vxy| ≤ M, so vxy can't span the middle region of a's.

(b) : L = {w ∈ {a, b, c}*. Again, the easiest thing to do is first to intersect L = {w ∈ {a, b, c}* : w has equal numbers of a's, b's, and c's} with a regular language. This time we construct L1 = L ∩ a*b*c*. L1 must be context free if L is. But L1 = a^n b^n c^n , which we've already proven is not context free. So L isn't either.

2. **Decide whether the following language is context-free or not, and state your reason:**
   a). $\{a^m b^n c^p: m = n \ or \ n = p \ or \ m = p\}$
   b). $\{a^m b^n c^p: m \neq n \ or \ n \neq p \ or \ m \neq p\}$
   c). $\{a^m b^n c^p: m = n \ and \ n = p \ and \ m = p\}$

   Sol: **(a)** It is a Context free as we can build a non deterministic PDA machine to accept it. It has three stacks. One for comparing n to m, then one for comparing m to p (skipping A's). and the last for comparing n to p (skipping b's)

   (b) : It is context-free, we can build a non deterministic PDA. It will have three stacks.

   (c) : It is context sensitive (not context free).
   We cannot design a non deterministic PDA for it. As we can only compare two variables.

We will push for one and pop for other (i.e., a, b), when it comes to c, it has freed up memory space and cant do anything for it.