**COT 4210 Homework #3: Section 1.3, 1.4 Solutions**

1) Give a regular expression generating the following languages:

$L_1$ = {w | w begins with a 1 and ends with a 0.}
$L_2$ = {w | w contains the substring 0101}
$L_3$ = {w | the length of w does not exceed 5}
$L_4$ = {w | every odd position of w is a 1}
$L_5$ = { w | w contains an odd number of 1s, or exactly 2 0s. }

**Solution**

$L_1$: $1(0 \cup 1)^*0$
$L_2$: $(0 \cup 1)^*0101(0 \cup 1)^*$
$L_3$: $(0 \cup 1 \cup \varepsilon)(0 \cup 1 \cup \varepsilon)(0 \cup 1 \cup \varepsilon)(0 \cup 1 \cup \varepsilon)(0 \cup 1 \cup \varepsilon)$
$L_4$: $\varepsilon \cup 1(01 \cup 11)^*(0 \cup 1 \cup \varepsilon)$
$L_5$: $(0^*10^*)(0^*10^*10^*)^* \cup 1^*01^*01^*$

2) Prove that the following languages are not regular via the Pumping Lemma:

$L_1$: $\{0^n1^n2^n \mid n \geq 0\}$
$L_2$: $\{ww \mid w \in \Sigma^*\}$
$L_3$: $\{a^{2^n} \mid n \geq 0\}$

**Solution**

(a) We will show that $L_1$ is not regular by showing that it does not satisfy the pumping lemma. Let p be the pumping length for $L_1$, assuming it's regular. Consider the string $s = 0^p1^p2^p$. the pumping lemma claims that there exists a way to rewrite s = xyz, where $|xy| \leq p$ and $|y| > 0$ such that $xy^*z$ is in $L_1$.

If we consider all ways to break down the string in this fashion, we must have $x = 0^i$ , $y = 0^j$ and $z = 0^{p-i-j}1^p2^p$, where $j \neq 0$.

Now, consider the string $xy^2z = 0^i0^{2j}0^{p-i-j}1^p2^p = 0^{p+j}1^p2^p$. Since $j \neq 0$, it follows that this string isn't in $L_1$. Thus, $L_1$ doesn't satisfy the pumping lemma and it follows that $L_1$ is not a regular language.

(b) We will show that $L_2$ is not regular by showing that it does not satisfy the pumping lemma. Let p be the pumping length for $L_2$, assuming it's regular. Consider the string $s = 0^p10^p1$. The pumping lemma claims that there exists a way to rewrite s = xyz, where $|xy| \leq p$ and $|y| > 0$ such that $xy^*z$ is in $L_2$.

If we consider all ways to break down the string in this fashion, we must have $x = 0^i$ , $y = 0^j$ and $z = 0^{p-i-j}10^p1$, where $j \neq 0$.

Now, consider the string $xy^2z = 0^i0^{2j}0^{p-i-j}10^p1 = 0^{p+j}10^p1$ Since $j \neq 0$, the left half of the string must be p+j zeros, followed by 1, followed by at least one 0. (Of course, if this length is odd, the string can't be in the language, so we only need to consider cases where this length is even.) This means the right half of the string would be some zeros followed by a single 1. It's impossible for this right half to equal the left half, since the former ends in 0 and the latter ends in 1. Thus, we have shown that $xy^2z$ is not in $L_2$. Since $L_2$ doesn't satisfy the pumping lemma, we can conclude that $L_2$ is NOT regular.

(c) We will show that $L_3$ is not regular by showing that it does not satisfy the pumping lemma. Let p be the pumping length for $L_1$, assuming it's regular. Consider the string $s = 0^{2^p}$. The pumping lemma claims that there exists a way to rewrite s = xyz, where $|xy| \leq p$ and $|y| > 0$ such that $xy^*z$ is in $L_3$.

If we consider all ways to break down the string in this fashion, we must have $x = 0^i$ , $y = 0^j$ and $z = 0^{2^p-i-j}$ , where $j \neq 0$.

Now, consider the string $xy^2z = 0^i0^{2j}0^{2^p-i-j} = 0^{2^p+j}$, where $0 < j \leq p$.

This string is not in the language. To see this, note that smallest string in the language with a length greater than the string s is $s' = 0^{2^{p+1}}$. To prove that our pumped string isn't in the language, we must show that its length is shorter than the length of s'. (We already know it's longer than the length of s, since $j > 0$.)

Note that for all non-negative integers $p < 2^p$. It follows that $2^p + j \leq 2^p + p < 2^p + 2^p = 2^{p+1}$. Thus the length of $xy^2z$ is longer than s, but shorter than s', proving that $xy^2z$ is NOT in $L_3$. Since $L_3$ doesn't satisfy the pumping lemma, it follows that $L_3$ is not regular.

3) Prove that $L = \{0^i1^j \mid j \neq i\}$ over the alphabet {0,1} is not regular via the pumping lemma.

**Solution**

Note: This question is more difficult than the previous one. The trick is that somehow, no matter what the substring y is, that there will be an integer number of times you can pump it so that the number of 0s and 1s becomes equal.

Assume to the contrary, that L is regular. If so, then L satisfies the pumping lemma for regular languages. Consider the string $s = 0^p1^{p!+p}$. This is a string in L that is of length p (the pumping length) or greater.

According to the pumping lemma, there exists a way to express s = xyz, with $|xy| <= p$ and $|y| > 0$, such that $xy^*z$ is in L.

Based on the given restrictions, we must have $x = 0^i$, $y = 0^j$, $z = 0^{p-i-j}1^{p!+p}$, where $i + j <= p$ and $j > 0$.

Our goal is to show that at least one string of the form $xy^*z$ is NOT in L. Thus, our goal should be to "pump" enough 0's so that the number of 0's must equal the number of 1's. Each time we pump the string, we add j number of 0's. In the original string, there are p! more 1's than 0s. Thus, we must add exactly p! number of 0's. We can do this by pumping exactly p!/j times. Since 1 <= j <= p, it is guaranteed that j divides evenly into p!.

Thus, we consider the string

$xy^{p!/j+1}z = xyy^{p!/j}z = 0^{p+p!/j*j}1^{p!+p} = 0^{p+p!}1^{p!+p} \notin L$, since the number of 0's and 1's in this string are equal. This contradicts the Pumping Lemma.

If follows that our assumption that L was regular must be faulty, thus L is not regular.

4) Using the result in class that if $L_1$ and $L_2$ are regular languages, then $L_1 \cap L_2$ is as well, prove the following assertion:

For languages $L_1$ and $L_2$, if $L_1 \cap L_2$ is NOT regular and $L_1$ is regular, then $L_2$ is not regular.

**<u>Solution</u>**
Let p, q and r be the following statements:

p: $L_1$ is a regular language.
q: $L_2$ is a regular language.
r: $L_1 \cap L_2$ is a regular language.

In class, we showed that $(p \wedge q) \rightarrow r$. We are asked to prove that $(\bar{r} \wedge p) \rightarrow \bar{q}$.

| | |
|---|---|
| $(p \wedge q) \rightarrow r$ | |
| $\overline{p \wedge q} \vee r,$ | Definition of implication |
| $\bar{p} \vee \bar{q} \vee r,$ | De Morgan's Law |
| $(r \vee \bar{p}) \vee \bar{q},$ | Commutative, Associative Laws |
| $\overline{\overline{r \vee \bar{p}}} \vee \bar{q},$ | Double Negation |
| $\overline{\bar{r} \wedge \bar{\bar{p}}} \vee \bar{q},$ | De Morgan's Law |
| $\overline{\bar{r} \wedge p} \vee \bar{q},$ | Double Negation |
| $(\bar{r} \wedge p) \rightarrow \bar{q}.$ | Definition of Implication |

In words, we can argue as follows:

We know that if both $L_1$ and $L_2$ are regular, that $L_1 \cap L_2$ is also. But, if we know that $L_1 \cap L_2$ is NOT regular, then the only way that can happen is if the statement, "both $L_1$ and $L_2$ are regular," is false. Thus, given that $L_1 \cap L_2$ is NOT regular, we can conclude that both $L_1$ and $L_2$ can't be regular. Furthermore, we are given that $L_1$ is regular. The only possible conclusion is that $L_2$ is NOT regular.

5) Utilizing the result proven in question #2, show that the following language, L, is not regular:

$$L = \{ a^i b^j c^k \mid i \geq 0, j \geq 0, k \geq 0, \text{ if } i = 1, \text{ then } j = k, \text{ otherwise, there are no restrictions of } j, k\}$$

**Solution**
Let $L_1 = ab^*c^*$ and $L_2 = \{ a^i b^j c^k \mid i \geq 0, j \geq 0, k \geq 0$, if $i = 1$, then $j = k$, otherwise, there are no restrictions of $j, k\}$.

It follows that $L_1 \cap L_2 = \{ab^i c^i \mid i \geq 0\}$. We can show that this language is not regular via the pumping lemma, using a nearly identical proof to the one shown in the book for $L = \{a^i b^i \mid i \geq 0\}$. Since $L_1$ is regular and $L_1 \cap L_2$ is not regular in this instance, using the result proved in question #2, we can conclude that $L_2$, the language presented in this question, is not regular.

6) Attempt to prove that L from question 3 is not regular utilizing the pumping lemma. What problem do you run into?

**Solution**
If we pick any string s that doesn't have one a in L, pumping s is trivial, since we can just add as many a's as we want. Thus, if we want any sort of hope to prove that L isn't regular via the pumping lemma, we must pick a string of the form $ab^n c^n$, for some value of n. An obvious choice would be $ab^p c^p$, where p is the pumping length. The problem here is that we could easily set $x = \varepsilon$, $y = a$, and $z = b^p c^p$, and all of the strings of the form $xy^*z$, would be in L, satisfying the pumping lemma. Thus, the general problem is that lots of strings are in this particular language L, and no matter which long string we pick in the language, there's always a choice for y that generates strings that are always in the language.