CS 181 Homework 2

Sriram Balachandran

TOTAL POINTS

87 / 100

QUESTION 1

Question 1 20 pts

1.1 1a 9 / 10

- **- 0 pts** Correct
- ✓ **- 1 pts Small Mistake**
- **- 3 pts** Flawed
- **- 5 pts** Major Flaws / Incomplete
- **- 8 pts** "I don't know"
- **- 10 pts** Incorrect / Missing

1.2 1b 8 / 10

- ✓ **+ 2 pts Not completely wrong**

Intuition

- ✓ **+ 2 pts Correct**
- **+ 1 pts** Not quite on track
- **+ 0 pts** Incorrect

Construction

- ✓ **+ 4 pts Correct**
- **+ 3 pts** Small Mistake
- **+ 2 pts** Flawed / Many Mistakes / Incomplete
- **+ 0 pts** Incorrect

Correctness Proof

- **+ 2 pts** Correct
- **+ 1 pts** Flawed / Incomplete
- ✓ **+ 0 pts Incorrect**

QUESTION 2

Question 2 40 pts

2.1 2a 10 / 10

- ✓ **- 0 pts Correct**
- **- 2 pts** Case: $$w \in b^m$$ for some $$m \ge 1$$ **partially correct**
- **- 2 pts** Case: $$w \in a^jb^p$$ for some $$j \ge 0$$ and $$p$$ prime **partially correct**
- **- 5 pts** Case: $$w \in b^m$$ for some $$m \ge 1$$ **incorrect**
- **- 5 pts** Case: $$w \in a^jb^p$$ for some $$j \ge 0$$ and $$p$$ prime **incorrect**
- **- 8 pts** "I don't know"
- **- 10 pts** No answer

2.2 2b 20 / 20

- ✓ **- 0 pts Correct**
- **- 5 pts** Missing explicit definitions for _a_ and _b_ in y
- **- 5 pts** No formal definition for loops / mention of pigeonhole principle
- **- 5 pts** No DFA/NFA construction
- **- 16 pts** "I don't know"
- **- 20 pts** No answer

2.3 2c 8 / 10

- **- 0 pts** Correct
- **- 4 pts** Incorrect partitioning for Pumping Lemma
- **- 3 pts** No use of generalized Pumping Lemma
- ✓ **- 2 pts No explicit equation to show # b's is composite**
- **- 4 pts** Picked an invalid word
- **- 8 pts** "I don't know"
- **- 10 pts** No answer

QUESTION 3

3 Question 3 32 / 40

- ✓ **+ 8 pts Not completely incorrect**

Intuition

- ✓ **+ 10 pts Correct**
- **+ 7 pts** Not quite on track

+ **0 pts** Incorrect

Construction of non-regular laguange

+ **15 pts** Correct
+ **13 pts** Small Mistake

✓ + **10 pts** **Missing Closure Argument**

+ **5 pts** Flawed
+ **0 pts** Incorrect

Correctness Proof

+ **7 pts** Correct
+ **5 pts** Incomplete

✓ + **4 pts** **Flawed**

+ **0 pts** Incorrect

1. **(20 points)** Let L_1 and L_2 be languages and define

$$\text{shuffle}(L_1, L_2) = \{x_1\, y_1\, x_2\, y_2\, \ldots\, x_n\, y_n \mid x_1\, \ldots\, x_n \in L_1, y_1\, \ldots\, y_n \in L_2\}.$$

For example, if $L_1 = \{1, 23, 45, 678\}$, *and* $L_2 = \{a, b, cd\}$ *then* $\text{shuffle}(L_1, L_2) = \{1a, 1b, 2c3d, 4c5d\}$.

(a) **(10 points)** Show that if the language L_1 is not regular and L_2 is any language then the languages $\text{shuffle}(L_1, L_2)$ and $\text{shuffle}(L_1, \overline{L_2})$ cannot both be regular.

(b) **(10 points)** Show that if L_1 and L_2 are regular languages then $\text{shuffle}(L_1, L_2)$ is regular.

Hint: For (a) recall closure properties of regular languages. **Answer:**

(a) Recall the alternating property that we proved in the previous homework assignment. We proved that for any regular language L, L_{alt} will always be a regular language, where L_{alt} is the language you get by leaving out the even-indexed letters for the words in L. Also recall that union is a closure property of regular languages.

We will prove that $\text{shuffle}(L_1, L_2)$ and $\text{shuffle}(L_1, \overline{L_2})$ cannot both be regular by contradiction. Let assume that both $\text{shuffle}(L_1, L_2)$ and $\text{shuffle}(L_1, \overline{L_2})$ are regular languages. Then:

$$L_3 = \text{shuffle}(L_1, L_2) \cup \text{shuffle}(L_1, \overline{L_2}) \tag{1}$$

Since union is closed over regular languages, we know that L_3 should be a regular language. Now recall that for any regular language L, L_{alt} will always be a regular language. Now let us apply the alternating property to L_3.

$$\text{shuffle}(L_1, L_2) = \{x_1\, y_1\, x_2\, y_2\, \ldots\, x_n\, y_n \mid x_1\, \ldots\, x_n \in L_1, y_1\, \ldots\, y_n \in L_2\}. \tag{2}$$
$$\text{alt}(\text{shuffle}(L_1, L_2)) = L_1 \tag{3}$$
$$\text{alt}(L_3) = L_1 \tag{4}$$

Under our assumption that both $\text{shuffle}(L_1, L_2)$ and $\text{shuffle}(L_1, \overline{L_2})$ are regular languages, we arrive at the conclusion that L_1 is also a regular language. However, this is a contradiction since we know L_1 to not be regular. Therefore we can see that if L_1 is not regular, then both $\text{shuffle}(L_1, L_2)$ and $\text{shuffle}(L_1, \overline{L_2})$ cannot be regular languages.

(b) Let $M(Q_m, \Sigma_m, \delta_m, q_{m0}, F_m)$ be the DFA that accepts L_1. Let $N = (Q_n, \Sigma_n, \delta_n, q_{n0}, F_n)$ be the DFA that accepts L_2. We can construct a DFA $P = (Q_p, \Sigma_p, \delta_p, q_{p0}, F_p)$ that accepts $\text{shuffle}(L_1, L_2)$ as follows:

$$Q_p = Q_m \times Q_n \times \{0, 1\} \tag{5}$$

$$\Sigma_p = \Sigma_m \cup \Sigma_n \tag{6}$$

$$q_{p0} = (q_{m0}, q_{n0}, 0) \tag{7}$$

The transition function will function as follows on a 3-state tuple. If the final entry in the 3-state tuple is 0, then we use M's transition function. If the final entry in the 3-state tuple

1.1 1a **9 / 10**

- **- 0 pts** Correct
- ✓ **- 1 pts Small Mistake**
- **- 3 pts** Flawed
- **- 5 pts** Major Flaws / Incomplete
- **- 8 pts** "I don't know"
- **- 10 pts** Incorrect / Missing

1. **(20 points)** Let L_1 and L_2 be languages and define

$$\text{shuffle}(L_1, L_2) = \{x_1\, y_1\, x_2\, y_2\, \ldots\, x_n\, y_n \mid x_1\, \ldots\, x_n \in L_1, y_1\, \ldots\, y_n \in L_2\}.$$

For example, if $L_1 = \{1, 23, 45, 678\}$, *and* $L_2 = \{a, b, cd\}$ *then* $\text{shuffle}(L_1, L_2) = \{1a, 1b, 2c3d, 4c5d\}$.

(a) **(10 points)** Show that if the language L_1 is not regular and L_2 is any language then the languages $\text{shuffle}(L_1, L_2)$ and $\text{shuffle}(L_1, \overline{L_2})$ cannot both be regular.

(b) **(10 points)** Show that if L_1 and L_2 are regular languages then $\text{shuffle}(L_1, L_2)$ is regular.

Hint: For (a) recall closure properties of regular languages. **Answer:**

(a) Recall the alternating property that we proved in the previous homework assignment. We proved that for any regular language L, L_{alt} will always be a regular language, where L_{alt} is the language you get by leaving out the even-indexed letters for the words in L. Also recall that union is a closure property of regular languages.

We will prove that $\text{shuffle}(L_1, L_2)$ and $\text{shuffle}(L_1, \overline{L_2})$ cannot both be regular by contradiction. Let assume that both $\text{shuffle}(L_1, L_2)$ and $\text{shuffle}(L_1, \overline{L_2})$ are regular languages. Then:

$$L_3 = \text{shuffle}(L_1, L_2) \cup \text{shuffle}(L_1, \overline{L_2}) \tag{1}$$

Since union is closed over regular languages, we know that L_3 should be a regular language. Now recall that for any regular language L, L_{alt} will always be a regular language. Now let us apply the alternating property to L_3.

$$\text{shuffle}(L_1, L_2) = \{x_1\, y_1\, x_2\, y_2\, \ldots\, x_n\, y_n \mid x_1\, \ldots\, x_n \in L_1, y_1\, \ldots\, y_n \in L_2\}. \tag{2}$$

$$\text{alt}(\text{shuffle}(L_1, L_2)) = L_1 \tag{3}$$

$$\text{alt}(L_3) = L_1 \tag{4}$$

Under our assumption that both $\text{shuffle}(L_1, L_2)$ and $\text{shuffle}(L_1, \overline{L_2})$ are regular languages, we arrive at the conclusion that L_1 is also a regular language. However, this is a contradiction since we know L_1 to not be regular. Therefore we can see that if L_1 is not regular, then both $\text{shuffle}(L_1, L_2)$ and $\text{shuffle}(L_1, \overline{L_2})$ cannot be regular languages.

(b) Let $M(Q_m, \Sigma_m, \delta_m, q_{m0}, F_m)$ be the DFA that accepts L_1. Let $N = (Q_n, \Sigma_n, \delta_n, q_{n0}, F_n)$ be the DFA that accepts L_2. We can construct a DFA $P = (Q_p, \Sigma_p, \delta_p, q_{p0}, F_p)$ that accepts $\text{shuffle}(L_1, L_2)$ as follows:

$$Q_p = Q_m \times Q_n \times \{0, 1\} \tag{5}$$

$$\Sigma_p = \Sigma_m \cup \Sigma_n \tag{6}$$

$$q_{p0} = (q_{m0}, q_{n0}, 0) \tag{7}$$

The transition function will function as follows on a 3-state tuple. If the final entry in the 3-state tuple is 0, then we use M's transition function. If the final entry in the 3-state tuple is 1, then we use N's transition function. Any transitions not covered by this function should lead to rejection.

$$\delta_p((m,n,0), a \in L_1) = (\delta_m(m,a), n, 1) \tag{8}$$

$$\delta_p((m,n,1), a \in L_2) = (m, \delta_n(n,a), 0) \tag{9}$$

By the way of construction of the shuffle language, we can see that words in the shuffle language must always finish with a letter of the second language. Thus our accepting states become as follows:

$$F_p = (m,n,0)|m \in F_m, n \in F_n \tag{10}$$

The intuition behind P is that we run both of the original DFAs in parallel and alternate back and forth between them. By way of the construction, we can see that our DFA will only accept words in shuffle(L_1, L_2). Since we can construct a DFA that accepts shuffle(L_1, L_2), shuffle(L_1, L_2) must be a regular language.

2. (**40 points**) In this problem we investigate the limits of the Pumping Lemma as it was stated in class and look for an alternative that remedies one of these shortcomings.

   (a) **(10 points)** Let L_1 be the language

   $$L_1 = \{a^i b^p \mid i \geq 0 \text{ and } p \text{ is a prime}\}.$$

   Prove that the language $L_2 = b^* \cup L_1$ satisfies the conditions of the Pumping Lemma. I.e. show that there exists a $q \in \mathbb{N}$ such that for every word $w \in L_2$ with $|w| \geq q$ we can write $w = xyz$ such that $|xy| \leq q$, $|y| > 0$, and for every $i \geq 0$, $xy^i z \in L_2$.

   (b) **(20 points)** Prove the following generalization of the Pumping Lemma:
   Let L be a regular language. There exists a $q \in \mathbb{N}$ such that for every $w \in L$ and every partition of w into $w = xyz$ with $|y| \geq q$ there are strings a, b, c such that $y = abc$, $|b| > 0$, and for all $i \geq 0$, $xab^i cz \in L$.

   (c) **(10 points)** Prove that the language L_2 is not regular.

Answer:

(a) Let $q = 2$. We can prove that L_2 satisfies the conditions of the Pumping Lemma as follows.
Case 1: $w \in L_1$

$$x = a, y = a, z = b^p \tag{11}$$

$$|xy| = 2 <= 2 \tag{12}$$

$$|y| = 1 > 0 \tag{13}$$

Case 2: $w \in b^*$

$$x = \epsilon, y = b, z = \epsilon \tag{14}$$

$$|xy| = 1 <= 2 \tag{15}$$

$$|y| = 1 > 0 \tag{16}$$

1.2 1b 8 / 10

✓ + **2 pts** **Not completely wrong**

Intuition

✓ + **2 pts** **Correct**

+ **1 pts** Not quite on track

+ **0 pts** Incorrect

Construction

✓ + **4 pts** **Correct**

+ **3 pts** Small Mistake

+ **2 pts** Flawed / Many Mistakes / Incomplete

+ **0 pts** Incorrect

Correctness Proof

+ **2 pts** Correct

+ **1 pts** Flawed / Incomplete

✓ + **0 pts** **Incorrect**

is 1, then we use N's transition function. Any transitions not covered by this function should lead to rejection.

$$\delta_p((m,n,0), a \in L_1) = (\delta_m(m,a), n, 1) \tag{8}$$

$$\delta_p((m,n,1), a \in L_2) = (m, \delta_n(n,a), 0) \tag{9}$$

By the way of construction of the shuffle language, we can see that words in the shuffle language must always finish with a letter of the second language. Thus our accepting states become as follows:

$$F_p = (m,n,0)|m \in F_m, n \in F_n \tag{10}$$

The intuition behind P is that we run both of the original DFAs in parallel and alternate back and forth between them. By way of the construction, we can see that our DFA will only accept words in shuffle(L_1, L_2). Since we can construct a DFA that accepts shuffle(L_1, L_2), shuffle(L_1, L_2) must be a regular language.

2. (**40 points**) In this problem we investigate the limits of the Pumping Lemma as it was stated in class and look for an alternative that remedies one of these shortcomings.

   (a) **(10 points)** Let L_1 be the language

   $$L_1 = \{a^i b^p \mid i \geq 0 \text{ and } p \text{ is a prime}\}.$$

   Prove that the language $L_2 = b^* \cup L_1$ satisfies the conditions of the Pumping Lemma. I.e. show that there exists a $q \in \mathbb{N}$ such that for every word $w \in L_2$ with $|w| \geq q$ we can write $w = xyz$ such that $|xy| \leq q$, $|y| > 0$, and for every $i \geq 0$, $xy^iz \in L_2$.

   (b) **(20 points)** Prove the following generalization of the Pumping Lemma:
   Let L be a regular language. There exists a $q \in \mathbb{N}$ such that for every $w \in L$ and every partition of w into $w = xyz$ with $|y| \geq q$ there are strings a, b, c such that $y = abc$, $|b| > 0$, and for all $i \geq 0$, $xab^icz \in L$.

   (c) **(10 points)** Prove that the language L_2 is not regular.

Answer:

(a) Let $q = 2$. We can prove that L_2 satisfies the conditions of the Pumping Lemma as follows.
Case 1: $w \in L_1$

$$x = a, y = a, z = b^p \tag{11}$$

$$|xy| = 2 <= 2 \tag{12}$$

$$|y| = 1 > 0 \tag{13}$$

Case 2: $w \in b^*$

$$x = \epsilon, y = b, z = \epsilon \tag{14}$$

$$|xy| = 1 <= 2 \tag{15}$$

$$|y| = 1 > 0 \tag{16}$$

2.1 2a **10 / 10**

✓ **- 0 pts** Correct

- 2 pts Case: $$w \in b^m$$ for some $$m \ge 1$$ **partially correct**

- 2 pts Case: $$w \in a^jb^p$$ for some $$j \ge 0$$ and $$p$$ prime **partially correct**

- 5 pts Case: $$w \in b^m$$ for some $$m \ge 1$$ **incorrect**

- 5 pts Case: $$w \in a^jb^p$$ for some $$j \ge 0$$ and $$p$$ prime **incorrect**

- 8 pts "I don't know"

- 10 pts No answer

(b) Let D be a DFA that accepts L with q states.

Let $w = w_1w_2w_3...w_n = xyz$, where $|y| >= q$. In order to process y, we would need to transition through $q+1$ states, since each letter in y represents a transition. Since there are only q states in D, by the pigeonhole principle, we know that at least one of these states must be repeated in the processing of y.

Let b be the substring of y that constitutes the state transition loop (causing states to be repeated), with length l such that $|y| >= l >= 1$. Then we can construct the following composition of $w = xab^icz$:

$$b = w_b...w_{b+l} \tag{17}$$

$$a = w_{y-start}...w_b \tag{18}$$

$$c = w_{b+l}...w_{y-end} \tag{19}$$

$$x = w_1...w_{y-start} \tag{20}$$

$$z = w_{y-end}...w_n \tag{21}$$

(c) Let $w \in L_2$, and let us assume L_2 to be regular. We can then apply the generalized pumping lemma as follows:

$$w = xyz|x = a^i, y = b^q, z = b^{p-q} \tag{22}$$

$$y = rst|r = b^d, s = b^e, t = b^f, d+e+f = q, q >= e >= 1 \tag{23}$$

Let $i = 1$. s can be repeated an arbitrary number of times and should still remain in L_2 as per the generalized pumping lemma. There is no guarantee that after repeated s an arbitrary number of times that the number of b's in w will be prime. By contradiction, L_2 is not regular.

3. (**40 points**) For a language L over alphabet Σ, we define

$$L_{\frac{1}{3}-\frac{1}{3}} = \{xz \in \Sigma^* \mid \exists y \in \Sigma^* \text{ with } |x| = |y| = |z| \text{ such that } xyz \in L\}.$$

For example, if $L = \{a, to, cat, math, solve, theory\}$, *then* $L_{\frac{1}{3}-\frac{1}{3}} = \{ct, thry\}$.
Prove that if L is regular, then $L_{\frac{1}{3}-\frac{1}{3}}$ need not be regular.
Hint: Consider the language 0^*21^* *and recall closure properties of regular languages*

Answer:

Let L be the regular language 0^*21^*. Then let us consider $L_{\frac{1}{3}-\frac{1}{3}}$:

Let $p \in \mathbb{N}$

$$\forall w \in L|w = 0^{2p-1}21^p \tag{24}$$

$$w_{\frac{1}{3}-\frac{1}{3}} = 0^p1^p \tag{25}$$

Therefore:

$$0^p1^p \subset L_{\frac{1}{3}-\frac{1}{3}} \tag{26}$$

Now let us apply the pumping lemma to the words in $L_{\frac{1}{3}-\frac{1}{3}}$'s sub-language 0^p1^p. There are 3 possible partitions of a word of in 0^p1^p in the form xy^iz.

2.2 2b **20 / 20**

✓ **- 0 pts** Correct

- 5 pts Missing explicit definitions for _a_ and _b_ in y

- 5 pts No formal definition for loops / mention of pigeonhole principle

- 5 pts No DFA/NFA construction

- 16 pts "I don't know"

- 20 pts No answer

(b) Let D be a DFA that accepts L with q states.

Let $w = w_1w_2w_3...w_n = xyz$, where $|y| >= q$. In order to process y, we would need to transition through $q+1$ states, since each letter in y represents a transition. Since there are only q states in D, by the pigeonhole principle, we know that at least one of these states must be repeated in the processing of y.

Let b be the substring of y that constitutes the state transition loop (causing states to be repeated), with length l such that $|y| >= l >= 1$. Then we can construct the following composition of $w = xab^icz$:

$$b = w_b...w_{b+l} \tag{17}$$
$$a = w_{y-start}...w_b \tag{18}$$
$$c = w_{b+l}...w_{y-end} \tag{19}$$
$$x = w_1...w_{y-start} \tag{20}$$
$$z = w_{y-end}...w_n \tag{21}$$

(c) Let $w \in L_2$, and let us assume L_2 to be regular. We can then apply the generalized pumping lemma as follows:

$$w = xyz|x = a^i, y = b^q, z = b^{p-q} \tag{22}$$
$$y = rst|r = b^d, s = b^e, t = b^f, d+e+f = q, q >= e >= 1 \tag{23}$$

Let $i = 1$. s can be repeated an arbitrary number of times and should still remain in L_2 as per the generalized pumping lemma. There is no guarantee that after repeated s an arbitrary number of times that the number of b's in w will be prime. By contradiction, L_2 is not regular.

3. (**40 points**) For a language L over alphabet Σ, we define

$$L_{\frac{1}{3}-\frac{1}{3}} = \{xz \in \Sigma^* \mid \exists y \in \Sigma^* \text{ with } |x| = |y| = |z| \text{ such that } xyz \in L\}.$$

For example, if $L = \{a, to, cat, math, solve, theory\}$*, then* $L_{\frac{1}{3}-\frac{1}{3}} = \{ct, thry\}$.
Prove that if L is regular, then $L_{\frac{1}{3}-\frac{1}{3}}$ need not be regular.
Hint: Consider the language 0^*21^* *and recall closure properties of regular languages*

Answer:

Let L be the regular language 0^*21^*. Then let us consider $L_{\frac{1}{3}-\frac{1}{3}}$:

Let $p \in \mathbb{N}$

$$\forall w \in L|w = 0^{2p-1}21^p \tag{24}$$
$$w_{\frac{1}{3}-\frac{1}{3}} = 0^p1^p \tag{25}$$

Therefore:

$$0^p1^p \subset L_{\frac{1}{3}-\frac{1}{3}} \tag{26}$$

Now let us apply the pumping lemma to the words in $L_{\frac{1}{3}-\frac{1}{3}}$'s sub-language 0^p1^p. There are 3 possible partitions of a word of in 0^p1^p in the form xy^iz.

2.3 2c **8 / 10**

- **- 0 pts** Correct
- **- 4 pts** Incorrect partitioning for Pumping Lemma
- **- 3 pts** No use of generalized Pumping Lemma
- ✓ **- 2 pts** **No explicit equation to show # b's is composite**
- **- 4 pts** Picked an invalid word
- **- 8 pts** "I don't know"
- **- 10 pts** No answer

(b) Let D be a DFA that accepts L with q states.

Let $w = w_1 w_2 w_3 ... w_n = xyz$, where $|y| >= q$. In order to process y, we would need to transition through $q+1$ states, since each letter in y represents a transition. Since there are only q states in D, by the pigeonhole principle, we know that at least one of these states must be repeated in the processing of y.

Let b be the substring of y that constitutes the state transition loop (causing states to be repeated), with length l such that $|y| >= l >= 1$. Then we can construct the following composition of $w = xab^i cz$:

$$b = w_b ... w_{b+l} \tag{17}$$
$$a = w_{y-start} ... w_b \tag{18}$$
$$c = w_{b+l} ... w_{y-end} \tag{19}$$
$$x = w_1 ... w_{y-start} \tag{20}$$
$$z = w_{y-end} ... w_n \tag{21}$$

(c) Let $w \in L_2$, and let us assume L_2 to be regular. We can then apply the generalized pumping lemma as follows:

$$w = xyz | x = a^i, y = b^q, z = b^{p-q} \tag{22}$$
$$y = rst | r = b^d, s = b^e, t = b^f, d + e + f = q, q >= e >= 1 \tag{23}$$

Let $i = 1$. s can be repeated an arbitrary number of times and should still remain in L_2 as per the generalized pumping lemma. There is no guarantee that after repeated s an arbitrary number of times that the number of b's in w will be prime. By contradiction, L_2 is not regular.

3. (**40 points**) For a language L over alphabet Σ, we define

$$L_{\frac{1}{3}-\frac{1}{3}} = \{xz \in \Sigma^* \mid \exists y \in \Sigma^* \text{ with } |x| = |y| = |z| \text{ such that } xyz \in L\}.$$

For example, if $L = \{a, to, cat, math, solve, theory\}$, *then* $L_{\frac{1}{3}-\frac{1}{3}} = \{ct, thry\}$.
Prove that if L is regular, then $L_{\frac{1}{3}-\frac{1}{3}}$ need not be regular.
Hint: Consider the language 0^*21^* *and recall closure properties of regular languages*

Answer:

Let L be the regular language 0^*21^*. Then let us consider $L_{\frac{1}{3}-\frac{1}{3}}$:

Let $p \in \mathbb{N}$

$$\forall w \in L | w = 0^{2p-1}21^p \tag{24}$$
$$w_{\frac{1}{3}-\frac{1}{3}} = 0^p 1^p \tag{25}$$

Therefore:

$$0^p 1^p \subset L_{\frac{1}{3}-\frac{1}{3}} \tag{26}$$

Now let us apply the pumping lemma to the words in $L_{\frac{1}{3}-\frac{1}{3}}$'s sub-language $0^p 1^p$. There are 3 possible partitions of a word of in $0^p 1^p$ in the form $xy^i z$.

(a)

$$y = 0^r | r <= p \tag{27}$$

$$x = 0^{p-r}, z = 1^p \tag{28}$$

It is trivial to see that if i is set to an arbitrarily high number, then the word will not be of form 0^p1^p anymore, meaning it will not be in the language and thus cause a contradiction. It is also not guaranteed to be in $L_{\frac{1}{3}-\frac{1}{3}}$.

(b)

$$y = 1^r | r <= p \tag{29}$$

$$x = 0^p, z = 1^{p-r} \tag{30}$$

It is trivial to see that if i is set to an arbitrarily high number, then the word will not be of form 0^p1^p anymore, meaning it will not be in the language and thus cause a contradiction. It is also not guaranteed to be in $L_{\frac{1}{3}-\frac{1}{3}}$.

(c)

$$y = 0^q1^r | r <= p, q <= p \tag{31}$$

$$x = 0^{p-q}, z = 1^{p-r} \tag{32}$$

It is trivial to see that if i is greater than 1, then the word will not be of form 0^p1^p anymore, meaning it will not be in the language and thus cause a contradiction. It will also not be in $L_{\frac{1}{3}-\frac{1}{3}}$.

Since there is no possible partition of the form xy^iz that will allow for $w \in 0^p1^p$ to remain in $L_{\frac{1}{3}-\frac{1}{3}}$, then by the pumping lemma, $L_{\frac{1}{3}-\frac{1}{3}}$ is not regular. This proves that even if L is regular, $L_{\frac{1}{3}-\frac{1}{3}}$ need not be regular.

3 Question 3 **32 / 40**

✓ **+ 8 pts** **Not completely incorrect**

Intuition

✓ **+ 10 pts** **Correct**

+ 7 pts Not quite on track

+ 0 pts Incorrect

Construction of non-regular laguange

+ 15 pts Correct

+ 13 pts Small Mistake

✓ **+ 10 pts** **Missing Closure Argument**

+ 5 pts Flawed

+ 0 pts Incorrect

Correctness Proof

+ 7 pts Correct

+ 5 pts Incomplete

✓ **+ 4 pts** **Flawed**

+ 0 pts Incorrect