Problem 1 (5 pts). Consider the following language

\[ \{ w \in \{0,1\}^* \mid \text{sum of } w \text{ mod } 3 = 0 \}, \tag{1} \]

where \( \epsilon \) is also included in the language. Recall in the second exam, for the case of mod 6 = 0, we have a 2-state PDA for the language. Now for (1) the 2-state PDA is

\[
\begin{align*}
1, 1 & \rightarrow 2 \\
0, \epsilon & \rightarrow \epsilon \\
1, \epsilon & \rightarrow 1 \\
0, \epsilon & \rightarrow \epsilon
\end{align*}
\]

Can this diagram be considered as a DPDA? If so, give the full table of \( \delta \). Otherwise, give the reasons why it is not a DPDA.

Solution.
Yes, the diagram above is a DPDA since for each state, input and top of stack, there is always only one possible transition (i.e. \( \forall q \in Q, a \in \Sigma, x \in \Gamma, \) exactly one of \( \delta(q, a, x), \delta(q, a, \epsilon), \delta(q, \epsilon, x), \delta(q, \epsilon, \epsilon) \) is not \( \emptyset \)). The table of \( \delta \) is:

<table>
<thead>
<tr>
<th>input</th>
<th>( \epsilon )</th>
<th>0</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>stack</td>
<td>( \epsilon, \epsilon )</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>( q_0 )</td>
<td>( \emptyset )</td>
<td>( (q_0, \epsilon) )</td>
<td>( \emptyset )</td>
<td>( (q_1, 1) )</td>
</tr>
<tr>
<td>( q_1 )</td>
<td>( \emptyset )</td>
<td>( (q_1, \epsilon) )</td>
<td>( \emptyset )</td>
<td>( \emptyset )</td>
</tr>
</tbody>
</table>

Comment mistakes: Your \( \Gamma \) does not need to have 0.

Problem 2 (25 pts). For the language (1), consider the standard TM (i.e., one-tape deterministic TM) to decide it. For subproblems (a) and (b), you are restricted to use

\[ \Sigma = \{0,1\} \text{ and } \Gamma = \Sigma \cup \{\sqcup\}. \]
(a) (10 pts) Give the state diagram and the formal definition of a TM that decides language (1). The total number of states (i.e. \( |Q| \), including \( q_{\text{accept}} \) and \( q_{\text{reject}} \)) should be less than or equal to 5. All states including \( q_{\text{reject}} \) should be drawn in the diagram.

(b) (10 pts) Give the state diagram of a two-tape TM that decides language (1) with only three states. Note that following the textbook, we allow \( S \) for the head movement.

(c) (5 pts) Simulate your machine in (b) on the input 1011. Show the configuration of the machine in each step.

Solution.

(a) We only need to simulate the DFA we already know using a TM. So, the state diagram can be

\[
\begin{align*}
  &q_0 \\ &\quad \rightarrow \text{R} \\ &\quad \rightarrow \text{R} \\
  &q_1 \\ &\quad \rightarrow \text{R} \\ &\quad \rightarrow \text{R} \\  &q_2 \\ &\quad \rightarrow \text{R} \\  &q_a \\ &\quad \rightarrow \text{R} \\ &\quad \rightarrow \text{R} \\  &q_r \\ &\quad \rightarrow \text{R} \\
\end{align*}
\]

The formal definition is

\[
Q = \{q_0, q_1, q_2, q_a, q_r\}
\]

\[
\Sigma = \{0, 1\}
\]

\[
\Gamma = \{\sqcup, 0, 1\}
\]

Start state: \( q_0 \)

Accept state: \( q_a \)

Reject state: \( q_r \)

\[
\delta : \\
\begin{array}{ccc}
  & 0 & 1 & \sqcup \\
 0 \rightarrow R & q_0 & (q_1, R) & (q_a, R) \\
1 \rightarrow R & q_1 & (q_2, R) & (q_r, R) \\
\sqcup \rightarrow R & q_2 & (q_0, R) & (q_r, R) \\
\end{array}
\]

(b) The idea is pretty similar to the PDA in problem 1. The machine reads the input tape just like a DFA (always going right). We use the second tape just like how we used the stack. Although we do not have symbol 2, we have 0, 1 and \( \sqcup \) that allow us to represent three different states on the second tape. We use \( \sqcup \) to represent the state that the sum mod 3 is 0, 0 for the case which sum is 1 and 1 for the case sum is 2. The diagram can be
Problem 3 (20 pts). Consider \( f(n) = \frac{2n + 80}{n^3} \).

(a) (10 pts) Prove \( f(n) = O\left(\frac{1}{n^2}\right) \)
by using the definition of big-O. That is, you must give some \( c \) and \( n_0 \) so the definition holds.

(b) (10 pts) Prove \( f(n) = o\left(\frac{1}{n}\right) \)
by using the definition of the limit, i.e., you cannot directly calculate the limit. Furthermore, for any given \( c \), please determine the smallest \( n_0 \).

Solution.

(a) Our target is finding \( c > 0 \) and \( n_0 > 0 \) such that
\[
\frac{2n + 80}{n^3} \leq \frac{c}{n^2}
\]
\[
\Rightarrow \frac{1}{n^2} \left( \frac{2n + 80}{n^3} \right) \geq 0
\]
\[
\Rightarrow \frac{cn - 2n - 80}{n^3} \geq 0
\]
\[
\Rightarrow \frac{(c - 2)n - 80}{n^3} \geq 0
\]
for all $n > n_0$. Thus, we can take $c = 3$
and $n_0 = 80$,
so that
\[
\frac{(c - 2)n - 80}{n^3} = \frac{n - 80}{n^3}
\]
is always greater than or equal to zero as $n > n_0 = 80$.

(b) The definition of the limit is
\[
\forall c > 0, \exists n_0, \forall n \geq n_0, \frac{2n + 80}{n^3} \leq \frac{1}{n},
\]
which is equivalent to
\[
\frac{1}{n} - \frac{2n + 80}{n^3} \geq 0 \\
\Rightarrow\frac{cn^2 - 2n - 80}{n^3} \geq 0 \\
\Rightarrow\frac{c(n - 1/c)^2 - 80 - 1/c}{n^3} \geq 0.
\]
We also know that the largest root of
\[
h(n) = c(n - \frac{1}{c})^2 - 80 - \frac{1}{c}
\]
is
\[
\tilde{n} = \frac{\sqrt{80c + 1} + 1}{c}
\]
and $h(n)$ is always greater than zero as $n > \tilde{n}$. Therefore, the smallest $n_0$ is equal to $[\tilde{n}]$, so that
\[
\frac{c(n - 1/c)^2 - 80 - 1/c}{n^3} \geq 0
\]
for all $n \geq n_0 = [\tilde{n}]$.
We can also list the $n_0$ (calculated above) corresponding to each $c$ in a table:

<table>
<thead>
<tr>
<th>$c$</th>
<th>$n_0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>[4, 5]</td>
<td>5</td>
</tr>
<tr>
<td>[6, 9]</td>
<td>4</td>
</tr>
<tr>
<td>[10,20]</td>
<td>3</td>
</tr>
<tr>
<td>[21,81]</td>
<td>2</td>
</tr>
<tr>
<td>[82, ∞]</td>
<td>1</td>
</tr>
</tbody>
</table>
Problem 4 (15 pts). Consider the following language
\[ \{w#w#\cdots#w \mid w \in \{0,1\}^*\}, \]
which includes \( \epsilon \).

(a) (10 pts) Give the diagram of a 2-tape TM (S is allowed) to recognize this language with \( \leq 6 \) states
(including \( q_{\text{accept}}, q_{\text{reject}} \)) and
\[ \Gamma = \{0,1,\sqcup,\#\} \]
by the following concepts

i) Copy \( w \) to the tape 2.
ii) If the machine reads \( \# \) in the tape 1, it moves the head of the tape 2 to the beginning.
iii) Continuously move the right for comparing contents in the tape 1 and the tape 2. If the machine
reads \( \# \) and \( \sqcup \) in the tape 1 and 2 respectively, go to ii). If the machine reads \( \sqcup \) and \( \sqcup \) in the
tape 1 and 2 respectively, it goes to the accept state.

The paths connected to \( q_{\text{reject}} \) can be omitted.

(b) (5 pts) Simulate the following input string

\[ 00\#00. \]

Solution.

(a) The diagram can be designed as
where we do not show the paths connected to $q_r$. From $q_1$ to $q_2$, we have a $\sqcup$ in tape 2 to indicate the beginning position. From $q_2$ to $q_2$, we copy $w$ to tape 2. From $q_2$ to $q_3$ and $q_3$ to $q_3$, we hold the head of the tape 1 on the $\#$ position and move the head of the tape 2 to the beginning position. From $q_3$ to $q_4$ and $q_4$ to $q_4$, we compare the contents in tape 1 and tape 2. If the machine reads $\#$ in tape 1 and the contents in tape 1 and tape 2 are the same, we will go back to $q_3$ from $q_4$. If the machine reads $\sqcup$ in both tape 1 and tape 2, and the contents in tape 1 and tape 2 are the same, the machine goes to $q_a$ from $q_4$. Moreover, we handle the empty word $\epsilon$ from $q_1$ to $q_a$, and handle the word $w$ without $\#$ from $q_2$ to $q_a$. 
Problem 5 (10 pts). Consider

\[
f(n) = n^3 \\
g(n) = 2^n
\]

(a) (5 pts) Is \( f(n) \times g(n) = O(2^n) \)?

(b) (5 pts) Is \( f(n) \times g(n) = 2^{O(n)} \)?

You cannot just answer yes or no. To have “\( = \)”, you need to prove the definition of big-O. To have “\( \neq \)” you must prove the opposite statement of the big-O definition.

Solution.

(a) No. Let us assume that \( f(n) \times g(n) = O(2^n) \).

By the definition, there exists \( c > 0 \) and \( n_0 \) such that

\[
c2^n - n^3 2^n \geq 0 \\
\Rightarrow (c - n^3)2^n \geq 0
\]

as \( n > n_0 \). However, for any \( c > 0 \) and any \( n_0 \), we consider

\[
n > \max(n_0, \sqrt[3]{c})
\]

such that

\[
(c - n_0^3)2^{n_0} < 0
\]

Thus,

\[
f(n) \times g(n) \neq O(2^n).
\]

Comment mistakes: The opposite statement has that for any \( c, n_0 \), we choose \( n \), so you cannot choose \( n_0 \).
(b) Yes. We have
\[ n^3 < 2^n \]
when \( n > 10 \). Thus,
\[ f(n) \times g(n) = n^3 2^n < 2^n \cdot 2^n = 2^{2n} \]
for all \( n > 10 \). By taking
\[ c = 2 \]
and
\[ n_0 = 10, \]
we have proved that
\[ f(n) \times g(n) = 2^{O(n)} \]
by the definition.

**Problem 6 (25 pts).** In problem 2(a), we used five states to construct the TM. We would like to prove that a four-state standard TM is not possible. However, the problem seems to be difficult, so we decide to simplify the language in (1) to
\[ L_3 = \{ w \in \{1\}^* \mid \text{sum of } w \text{ mod } 3 = 0 \}, \]
where \( \epsilon \in L_3 \) and prove that no four-state TM can decide it. We give the proof below, but three parts are taken out. Your task is to fill those parts.

For the proof, we (and you) may use the following lemmas. Due to the interest of time, you can skip reading their proofs.

**Lemma 1.** Let \( M \) be a TM that recognizes \( L_3 \) and \( w \) be a string that is accepted or rejected by \( M \) (including rejection by an infinity loop). Then, before accepting or rejecting \( w \), \( M \) must have read the \( \sqcup \) at the end of the input. Moreover, \( M \) takes at least \( |w| \) steps before accepting or rejecting \( w \).

**Proof of Lemma 1.** Suppose that \( M \) is a TM that recognizes \( L_3 \), and \( M \) accepts or rejects some string \( w \) without ever reading the \( \sqcup \) at the end of the input. Then, \( M \) must make the same decision of \( w \) for the strings \( w1 \) and \( w11 \). As a result, \( M \) does not recognize \( L_3 \) because it is not possible that \( w, w1 \) and \( w11 \) are all in or not in \( L_3 \). Therefore, \( M \) must have read the \( \sqcup \) at some point. Further, to read up to this \( \sqcup \) at the end of the input, \( M \) needs at least \( |w| \) steps to move the head there. \( \square \)

**Lemma 2.** There does not exist a TM with
\[ |Q| = 3, \Sigma = \{1\}, \Gamma = \{1, \sqcup\} \]
that recognizes \( L_3 \).

**Proof of Lemma 2.** Suppose there exists a machine \( M \) that recognize \( L_3 \) with
\[ Q = \{q_0, q_a, q_r\}, \]
where \( q_0, q_a \) and \( q_r \) are the start, accept and reject states, respectively. A TM must terminate when it goes to \( q_a \) or \( q_r \). Thus, there are no transitions starting from \( q_a \) and \( q_r \), so we only need to discuss the transitions from \( q_0 \) to \( q_a \), \( q_0 \) to \( q_r \) and \( q_0 \) to \( q_0 \). Furthermore, because
\[ |\Gamma| = 2, \]
we know that there are exactly two transitions from $q_0$ to some states by respectively handling input characters 1 and $\sqcup$.

To accept the strings in $L_3$, there must be a transition

$$q_0 \xrightarrow{x} q_a,$$

(3)

where $x \in \Gamma$. If $x$ is 1, $M$ would accept 1, which is a contradiction. Therefore, (3) must be

$$q_0 \xrightarrow{\sqcup} q_a.$$

(4)

By (2) and (4), we only need to discuss two cases for the other transition:

Case 1:

$$q_0 \xrightarrow{1} q_r$$

Case 2:

$$q_0 \xrightarrow{1, A} q_0$$

(5)

where $A \in \{R, L\}$.

For Case 1, $M$ rejects 111, which is a contradiction. For Case 2, if $A$ is $L$, then $M$ can only read the first cell of the tape. Because $M$ recognizes $L_3$, $M$ would accept 111 without ever touching the $\sqcup$ at the end of input, which is a contradiction by Lemma 1. Therefore, $A$ should be $R$. For the input string 1, after one transition by (5), the head points to a $\sqcup$, so $M$ takes (4) to accept the string. Then we get a contradiction.

Now, we can begin to prove there are no four-state TM that decides $L_3$. Suppose there is a machine $M$ that decides $L_3$ with

$$\Gamma = \{1, \sqcup\}, Q = \{q_0, q_1, q_a, q_r\},$$

where $q_0$, $q_a$ and $q_r$ are the start, accept and reject states, respectively. Because there are no transitions starting from $q_a$ and $q_r$, we only have to consider 4 transitions in total (two of the transitions start from $q_0$, the other two start from $q_1$). Hence, we have the following discussion:

Discussion 1: The first transition must be a transition $? \rightarrow q_r$. Because $M$ is a decider, $M$ does not rely on infinite loops to reject a string. Therefore, $M$ must reject some string by transitioning to $q_r$.

Discussion 2: The second transition must be $? \rightarrow q_a$, since $M$ must accept some strings.

Discussion 3: The third transition must be $q_0 \rightarrow q_1$. If such a transition does not exist, $q_1$ is never reached and therefore $M$ is equivalent to a three-state TM. By Lemma 2, $M$ cannot decide $L_3$, so there is a contradiction.

Discussion 4: Following Discussion 1-3, the last transition can neither go to $q_a$ nor $q_r$. Otherwise, the machine can only transition at most once before deciding. For example, assume that the diagram is like
Then, after one step ($q_0$ to $q_1$), the TM terminates by going to $q_a$ or $q_r$. By the assumption that $M$ decides $L_3$, Lemma 1 implies that $M$ accepts 111 with at least three steps. Thus, there is a contradiction.

From Discussion 1-4, we know that there are exactly one transition to $q_a$ and exactly one transition to $q_r$. We can then separate the discussion into two cases:

(a) Both transitions to $q_a$ and $q_r$ read $\sqcup$

Because the language includes $\epsilon$, the machine should have

$$q_0 \xrightarrow{\sqcup} q_a$$

instead of

$$q_0 \xrightarrow{\text{⋯}} q_r.$$  

Since the transition to $q_r$ also reads $\sqcup$, the determinism of TM and (6) imply that a transition must be

$$q_1 \xrightarrow{\sqcup} q_r.$$  

Also, by Discussion 3, there must be a transition from $q_0$ to $q_1$. Now, the diagram looks like:

Next, when reading 1 in $q_1$, it should go back to $q_0$. Otherwise, $M$ either loops at $q_1$ or goes to $q_r$. In either case, $M$ does not accept 111, which is a contradiction. Thus, the diagram should look like
where \( x, y \in \Gamma \) and \( A, B \in \{ L, R \} \).

(10 pts) **Please finish the proof of this case by checking all possibilities of \( A \) and \( B \).** Then, conclude that all of them lead to contradiction. (Hint: Using the lemmas properly should make things easier)

(b) At least one of the two transitions to \( q_a \) and \( q_r \) reads 1.

By the fact that one of the two deciding transitions reads 1 and \( M \) is a decider, there must be a string \( w \) that is decided when reading 1. For this \( w \), Lemma 1 implies that \( M \) must have read the \( \sqcup \) at the end of input before acceptance or rejection. If after reading this \( \sqcup \), the head of \( M \) only goes right (i.e., all subsequent transitions are \( ? \rightarrow ?, R \)), \( M \) does not have a chance to read a 1 afterward. However, we already knew that for this \( w \), the last transition before reaching \( q_a \) or \( q_r \) reads a 1. Therefore, we conclude that

“for the two transitions going to neither \( q_a \) nor \( q_r \), one must move left.” (7)

From Discussion 1-4, there are four transitions in total. That is, \( q_0 \rightarrow q_1 \), \( ? \rightarrow q_a \), \( ? \rightarrow q_r \) and the last one goes to neither \( q_a \) nor \( q_r \). Therefore, there are only 4 possibilities of the last edge:

(i) \( q_0 \rightarrow q_0 \)
(ii) \( q_0 \rightarrow q_1 \)
(iii) \( q_1 \rightarrow q_0 \)
(iv) \( q_1 \rightarrow q_1 \)

We discuss each of them separately. Note that in each corresponding diagram, there are two transitions without showing the target states (e.g., two from \( q_1 \) to nowhere in the first figure). They are the transitions to \( q_a \) and \( q_r \). In all cases, the assignment of the target state \( q_a \) or \( q_r \) does not affect our discussion, so they do not need to be specified.

(i)

\[ ? \rightarrow x, A \]

\[ \rightarrow q_0 \quad ? \rightarrow y, B \quad \rightarrow q_1 \]

According to (7), at least one of \( A \) and \( B \) must be \( L \). We separately consider the following cases:

Case 1. \( A = B = L \): \( M \) can not read more than one input character since it does not have any transition that moves right. By the assumptions that \( M \) recognizes \( L_3 \), \( M \) accepts 111 without ever reading the \( \sqcup \) after of input. This contradicts Lemma 1.

Case 2. \( A = L \) and \( B = R \): In this case, when \( M \) reaches \( q_1 \) before deciding, at most two input characters have been read. By the same reason as in case b(i)1, a contradiction occurs by Lemma 1.

Case 3. \( A = R \) and \( B = L \): Let \( w \) be any input string such that \( |w| \geq 3 \). If \( M \) does not use the transition \( q_0 \rightarrow q_0 \), it must decide \( w \) in only two steps, which contradicts Lemma 1 since \( |w| \geq 3 \). Because \( A = R, B = L \) and that \( q_0 \rightarrow q_0 \) is used at least once, the sequence of transitions must be:

\[ q_0 \rightarrow q_0 \rightarrow \cdots \rightarrow q_0 \xrightarrow{? \rightarrow x, R} q_0 \xrightarrow{? \rightarrow y, L} q_1 \xrightarrow{x} \]
This implies that either $M$ rejects both of 111 and 1111, or accepts both of 111 and 1111. Thus, there is a contradiction.

(ii)

This machine makes decision after reading at most the first two tape cells. Therefore, it must accept 111 before reading the ⊥ next to the input. This contradicts Lemma 1.

(iii) (5 pts) Please finish the proof for the case where the last edge is $q_1 \rightarrow q_0$.

(iv) (10 pts) Please finish the proof for the case where the last edge is $q_1 \rightarrow q_1$.

All of the four cases lead to contradictions. Thus, there are no 4-state TM that decides $L_3$ in this case.

Solution.

(a) We can list all possibilities of $A$ and $B$ and give a reason for their contradictions in the following table:

<table>
<thead>
<tr>
<th>$A, B$</th>
<th>contradiction</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A = B = L$</td>
<td>can only read the first input</td>
</tr>
<tr>
<td>$A = L, B = R$</td>
<td>can only read the first two input</td>
</tr>
<tr>
<td>$A = R, B = L$</td>
<td>can only read the first two input</td>
</tr>
<tr>
<td>$A = B = R$</td>
<td>no matter of what $x, y$ is, it accepts 11</td>
</tr>
</tbody>
</table>

Because the fourth case in the table cannot happen, $M$ satisfies one of the first three cases in the table, which indicates that $M$ only reads the first or two cells. By the assumption that $M$ decides $L_3$, $M$ accepts 111 without reading the ⊥ at the end of 111. This is a contradiction to Lemma 1. Therefore, no 4-state standard TM can decide $L_3$ in this case.

**Common mistake:** Some directly wrote: If $A = L, B = R$, then 11 cause an infinite loop. You cannot have such result without discussing $x$ and $y$. Indeed, if $y = ⊥$, then 11 is accepted.

(b)

(iii)

From (7), one of the two transitions between $q_0$ and $q_1$ must move left. Then, the machine can not read more than two tape cells. By the same explanation as in the case (bii), this situation is not possible.

**Common mistake:** If your figure is wrong, then it is impossible to have a correct proof. For example, some have:
However, at $q_1$ you should only have two transitions going out.

(iv)

![Diagram](image)

Due to the determinism of TM, in this diagram we have

$$\{x, y\} = \{1, \sqcup\} \text{ and } x \neq y.$$  

Then, we consider the following possible situations:

Case 1. $x = 1, y = \sqcup$: Because $M$ is a decider, it must not loop forever on any input. Therefore, $M$ make the same decision for all input strings that start with 1. That is, either 111 and 11 are both rejected, or the machine accepts both 111 and 11. This situation is thus impossible.

Case 2. $x = \sqcup, y = 1$: The machine immediately goes to $q_a$ or $q_r$ for any input that starts with 1. Thus, it makes the same decision for all input that starts with 1. By the same reason as the above case, this situation is not possible.

Alternative solution:

![Diagram](image)

By (7), at least one of $A$ and $B$ is $L$. Then, we can discuss three cases:

i. If $A = B = L$, the machine only read the first tape cell. By the assumption that $M$ decides $L_3$, $M$ accepts 111 without reading $\sqcup$ at the end of input. We have a contradiction to Lemma 1.

ii. If $A = R$ and $B = L$, the machine only read the first two tape cells. By the same reasons as in b(iv)i, we have a contradiction.

iii. Suppose $A = L$ and $B = R$. Consider any input string $w$ such that $|w| \geq 2$. If $q_1 \rightarrow q_1$ is not used when deciding $w$, $M$ can not read the $\sqcup$ at the end of input, which contradict Lemma 1. By the fact that $M$ is a decider and $q_1 \rightarrow q_1$ must be used when processing long inputs, $M$ would always transition first to $q_1$ and end up in the same final state (since there is only one transition out of $q_1$). This implies $M$ either accepts both of 111 and 1111, or reject both of them, a contradiction.