Try to keep your answers succinct.

1. (10 points) What is the language accepted by the following Turing machine? (Recall that “1/BR” means on reading a 1, write a Blank and move Right.)

   ![Turing Machine Diagram](image)

   q1  1/BR  B/BR  q2  1/BR

2. Show that the following are true:

   (a) (7 points) If \( L \) is r.e., and \( L \cong \overline{L} \), then \( L \) is recursive.

   (b) (7 points) If \( L \) is recursive, then \( L \cong 0^*1^* \).

3. In this problem, we show that the following language is r.e., but not recursive:

   \( L_{\text{steps}} = \{ \langle M \rangle : \text{there exist distinct strings } w_1 \text{ and } w_2 \text{ such that } M \text{ accepts } w_1 \text{ and } w_2 \text{ in the same number of steps} \} \)

   (a) (8 points) Show that \( L_{\text{steps}} \) is r.e. (Hint: It may help to write language \( L \) as \( \{ \langle M \rangle : \exists w_1, w_2, t \text{ s.t. \ldots} \} \).)

   (b) (6 points) Keeping in mind part (a), explain why we would not try to use each of the following possible reductions to show that \( L_{\text{steps}} \) is not recursive.

   i. Why won’t \( 0^*1^* \cong L_{\text{steps}} \) help?

   ii. Why won’t \( L_{\text{steps}} \cong L_u \) help?

   iii. Why won’t \( \overline{L_u} \cong L_{\text{steps}} \) help?

   (c) (8 points) Prove that \( L_{\text{steps}} \) is not recursive by showing that \( L_u \cong L_{\text{steps}} \). (If you cannot do this part, at least be clear about what “you want” from your reduction.)

   (d) (4 points) Is the complement of \( L_{\text{steps}} \) r.e.? Why or why not?

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>/10</td>
</tr>
<tr>
<td>2.</td>
<td>/14</td>
</tr>
<tr>
<td>3.</td>
<td>/26</td>
</tr>
<tr>
<td>Total</td>
<td>/50</td>
</tr>
</tbody>
</table>