**Objective**
Develop steps to sort a row of cards, using the rules below.

**Tips**
- Start with only 3 or 4 cards and work up to 8.
- Switch roles frequently.
- Practice with the cards face up first.
- Test your work and even try to “break” your solution.
- Stick with it! There are many possible solutions. The point is to find yours!

**Rules**

<table>
<thead>
<tr>
<th>Sorter</th>
<th>Pointer</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Can pick up, put down, and swap cards in any order they wish.</td>
<td>1. When shown two cards can point to the bigger one.</td>
</tr>
<tr>
<td>2. Shows cards to pointer without looking at them.</td>
<td>2. For a tie point to either.</td>
</tr>
<tr>
<td>3. Only one card in each hand at a time.</td>
<td>3. No other communication is allowed.</td>
</tr>
<tr>
<td>4. Cards go back into empty spots on the table.</td>
<td></td>
</tr>
</tbody>
</table>

**Your Steps**
Once you’ve developed your steps for processing cards, write the steps of your way of processing the cards in the space below. Feel free to draw pictures, number steps, or do anything else you think would be helpful.

**Share and Test**
Present your steps to another group. Make sure both groups get a chance to share and test their ideas. Here’s some tests to consider:
- Use different numbers of cards
- Cards are in reverse order
- Cards are already in order
- Cards are nearly in order
**Iterate**
Based on your tests or ideas from another group, update your steps.

**Processing Uses Algorithms**
Processing is anything a computer does to turn input information into output information. Humans can process information, but usually they're making lots of assumptions or mental leaps to do it. When computers process information, they use **algorithms**, or sets of instructions, that will always turn an input into a desired output. The steps you just created are an algorithm to sort cards.

**Reflection**

1. What step of the problem-solving process did you think was most important in this activity? ____________________

2. Why?

________________________________________________________
________________________________________________________

3. Why would someone create an algorithm if they already know how to solve a problem by hand?

________________________________________________________
________________________________________________________
________________________________________________________