Activity - Minimum Card Algorithm

Algorithms with Lists

Once you know how to write programs that can store data in lists, the next step is to think about how you might write code, or develop algorithms, to process those lists to find or do interesting things with the data.

We often get started thinking about algorithms by trying to rigorously act them out ourselves. Because we know something about how computer programs work, we can keep the limitations of a computer in mind when we are “acting as the machine.”

In this activity, you’ll design an algorithm to find the smallest item in a list. Obviously, if we were really writing instructions for a person, we could simply tell them: “find the smallest item in a list.” But that won’t work for a computer. We need to describe the process that a person must go through when they are finding the smallest item. What are they really doing?

Setup and Rules:

● We’ll use playing cards face down on the table to represent a list of items. Start with 8 cards face down in a row. Any card on the table must be face down.

● When acting as the machine, you can pick up a card with either hand, but you can only hold one card at a time in each hand.

● You can also compare the values on playing cards to determine which one is greater than the other (based on its face value).

● You can put a card back down on the table (face down), but once a card is face down on the table, you cannot remember (or memorize) its value or position in the list.

Task:

Write an algorithm to find the card with the lowest value in the row of cards.

● Goal: The algorithm must have a clear end to it. The last instruction should be to say: “I found it!” and hold up the card with the lowest value.

● The algorithm should be written so that it would theoretically work for any number of cards (1 or 1 million).

● Write your algorithm out on paper as a clear list of instructions in “pseudocode.” Your instructions can refer to cards, and a person’s hands, etc., but you must give a systematic way for finding the smallest card.
<table>
<thead>
<tr>
<th>Algorithm To Find Minimum Card</th>
</tr>
</thead>
</table>