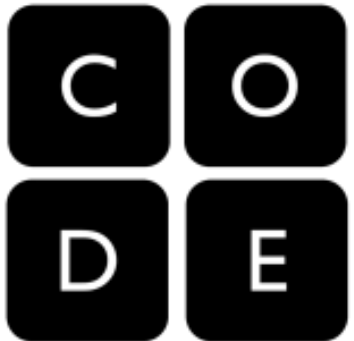
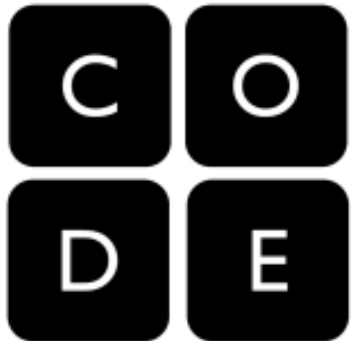


Creativity in Algorithms



Opening thoughts:

One thing about algorithms is that once you know a few, and know how they work, you can combine them (or slightly modify them) to solve new problems. **Creativity** in algorithms comes from figuring out clever ways to solve problems by developing a process that could be executed by a machine.

- We study algorithms and care about them because in programming the techniques and mechanics of certain processes come up over and over and over again. So it's worth knowing a few so you don't have to reinvent the wheel.
- For example, you just wrote an algorithm to find the smallest card in a row of cards. Is it hard to modify that exact same strategy to find the max card?
- Today we'll challenge you with a few problems that will require you to ***get creative!***

Human Machine Language - Part 2 (pass out Activity Guide):

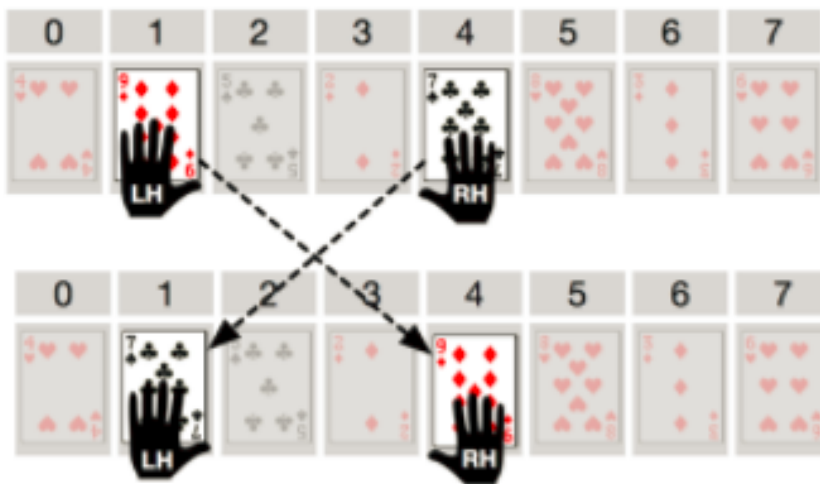
We're going to add one command to the Human Machine Language called **SWAP** - see description below. All of the other commands are still available to you. So, there are 6 commands total in the language now.

1st: Review the new command

2nd: In pairs, attempt the example problem

SWAP

Swap the positions of the cards currently being touched by the left and right hands. After a swap the cards have changed positions but hands return to original position.



The human machine action is: pick up the cards, exchange the cards in hand, and return hands to original position in the list with the other card.

Human Machine Language Reference

SHIFT TO THE

MOVE TO POSITION

JUMP TO LINE

JUMP TO LINE IF

SWAP

STOP

Human Machine Language - Part 2:


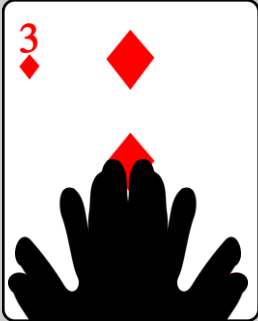

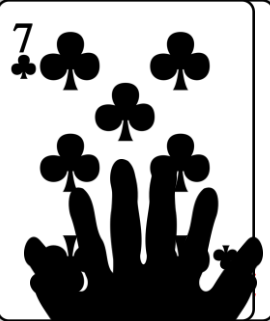
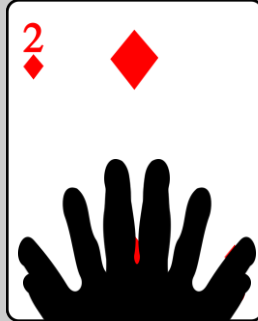
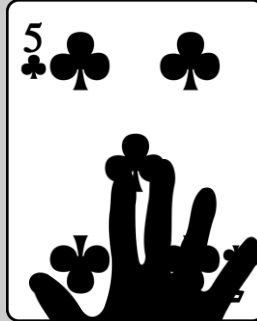
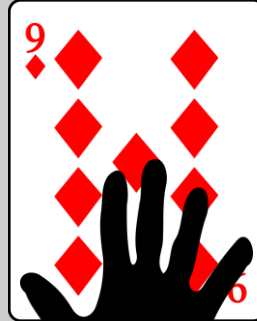
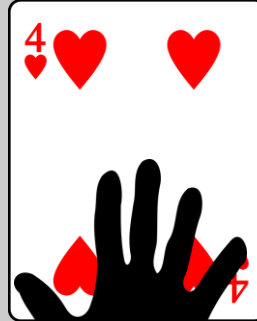
Try an example with Swap

Trace the program below with a partner and describe what it does.

1	MOVE RH TO POSITION 7
2	SWAP
3	SHIFT LH TO THE R
4	SHIFT RH TO THE L
5	JUMP TO LINE 2 IF RHPos gt LHPos
6	STOP

What does this program do?

SWAP

0	1	2	3	4	5	6	7
							

2	
3	
4	
5	
6	

Human Machine Language - Part 2:

Try an example with Swap

Trace the program below with a partner and describe what it does.

1	MOVE RH TO POSITION 7
2	SWAP
3	SHIFT LH TO THE R
4	SHIFT RH TO THE L
5	JUMP TO LINE 2 IF RHPos gt LHPos
6	STOP

What does this program do?

- END STATE: the order of the cards has been reversed
- It does this by first moving the right hand to the end of the list, then shifting each hand progressively toward the middle of the row, swapping cards each time.
- The program stops once the

Human Machine Language - Part 2

CHALLENGE: *Minimum to the Front*

Using only the Human Machine Language design an algorithm to find the smallest card and move it to the front of the list (position 0). All of the other cards *must remain in their original relative ordering*.

END STATE: When the program stops, the smallest card should be in position 0. The ending positions of the hands do not matter, the ending positions of the other cards do not matter. As a *challenge*: try to move the min-to-front and have all other cards be in their original relative ordering.

Cards BEFORE:

0	1	2	3	4	5	6	7
9	4	5	2	7	8	3	6

Cards AFTER (may not be in this order)

0	1	2	3	4	5	6	7
2	9	4	5	7	8	3	6

Human Machine Language - Part 2

*Minimum to the Front - **Solutions Review***

1st: Swap your program with another pair, and see if it works

2nd: 2 or more groups share their approach with the class

Wrap Up

IN YOUR NOTES, get down the following definitions. (*Where did you see these pop-up in our Human Machine Language?*)

- **Iterate** - To repeat in order to achieve, or get closer to, a desired goal.
- **Selection** - A generic term for a type of programming statement (usually an if-statement) that uses a Boolean condition to determine, or select, whether or not to run a certain block of statements.
- **Sequencing** - Putting commands in correct order so computers can read the commands.

Wrap Up

- Learning to program is really learning how to think in terms of algorithms and processes. And it can be really fun and addicting. It also can make you feel like you have magical powers.
- In the next lesson we'll start writing programs that a real machine (not a human machine) will execute. But since programming is really about thinking and problem solving your "human machine" skills will come in handy - programming is also about reasoning about what a computer can and cannot do, and what the given language you're using lets you and doesn't let you do.
- If you didn't want to learn how to program before, perhaps this video will change your mind...

Wrap Up (10-minutes)

