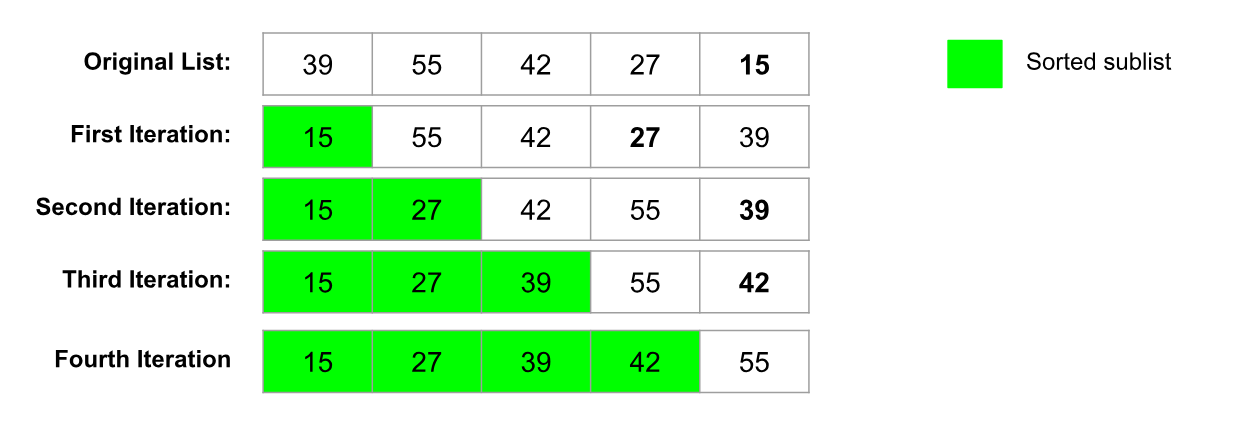
**Algorithm 1**

Observe the following sort and answer the questions below.



Observe the Original list. What is the smallest value? **15**

What happened to that number in the First iteration of the Sort? **The smallest value of the list got moved to the first position, swapping with the original first position.**

Observe the Second iteration.

What is the smallest number within the unsorted portion? **39**

What happens to that number on the Second iteration? **It gets moved to the 3rd position of the list, putting it in the right spot.**

Is there a pattern to the sorting? **Yes, the smallest item gets moved to the next position, based on the number of iterations.**

As a group, create a definition for this algorithm.

**Find the smallest item in the unsorted list, and move it to the beginning of the unsorted list, regarding it then as part of the sorted list. Or, take the smallest item and move it to the last position of the sorted list.**

Compare your definition to the Algorithms we discussed in class. Which algorithm do you think this is? **This is a selection sort.**

**Sort the Cards:** *Mix your cards up and place them face down on the table, sort them according to the algorithm by following the steps below.*

*Pick 2 cards up at a time to compare values. Keep track of the number of comparisons by adding a hash mark in the space provided below.*

**Step 1:** Find the smallest card in the unsorted list: Choose the first card in the list. Compare it to the next card in the list. Keep the smallest card and compare it to the next card in the list. Repeat this process until the end of the list. You now have the smallest card.

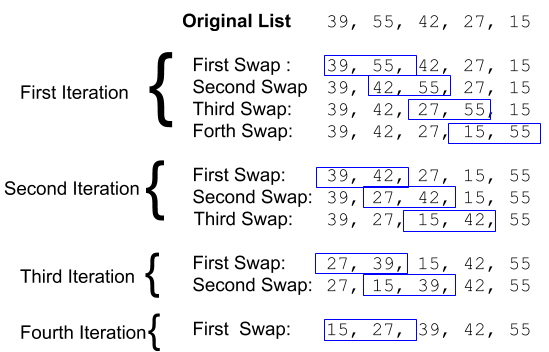
**Step 2:** Swap the smallest card places with the front item in the list, it now in the sorted so do not include this card again in your search for the smallest card.

**Step 3:** Repeat the process with the unsorted portion of the list

**Number of comparisons:** 45 - 9, 8, 7, 6, 5, 4, 3, 2, 1

**Algorithm 2**

Observe the following sort and answer the questions below.



Observe the first iteration of the algorithm.

Observe the First Swap. Is there any change to the list? **No there isn’t**

Observe the second swap. Was there a change in the list? **Yes, 42 & 55 swapped**

Observe the third swap. Was there a change in the list? **Yes, 27 & 55 swapped**

Observe the pattern for the rest of the iterations.

As a group, create a definition for this algorithm.

**For each item in the list, compare it to the next. If it is larger than the next item, swap them.**

Compare your definition to the Algorithms we discussed in class. Which algorithm do you think this is? **Bubble sort**

**Sort the Cards:** *Mix your cards up and place them face down in a line on the table, sort them according to the algorithm by following the steps below.*

*Pick 2 cards up at a time to compare values. Keep track of the number of comparisons by adding a hash mark in the space provided below.*

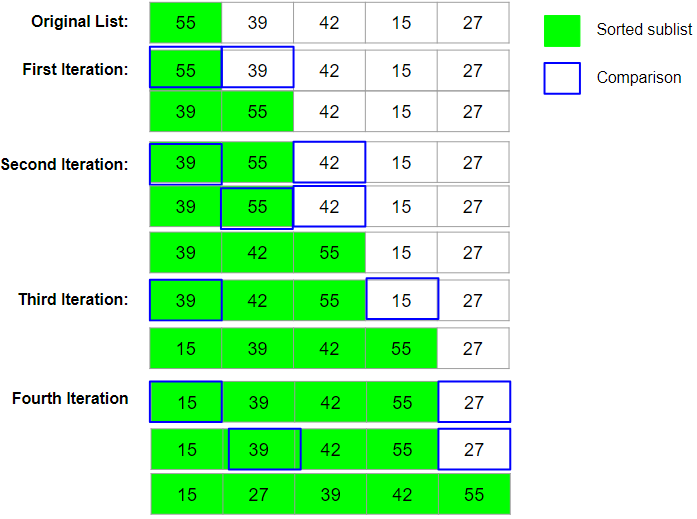
Compare the first two cards. Rearrange the cards if they are not in order. Compare the second card with the third, rearranging as needed. Repeat the process till the last card. The last card is now sorted.

Repeat this process again starting with the first two cards. Do not include the sorted cards at the end.

Number of comparisons: **45** (Again)

**Algorithm 3**

Observe the following sort and answer the questions below.



How many values are in the sorted sublist in the original List? **1**

What is the pattern where numbers are inserted into the sublist? **It compares to the items in the list in order and then, when the item it compares against is larger than it, it goes there.**

As a group, create a definition for this algorithm. **Add the first item of your list to a separate sorted list. For each item in your unsorted list, compare it to each of the items in the sorted list in order. When the value of the item you compare against is larger than your current value, insert your value before the larger item.**

Compare your definition to the Algorithms we discussed in class. Which algorithm do you think this is? **Insertion sort.**

**Sort the Cards:** *Mix your cards up and place them face down in a line on the table, sort them according to the algorithm by following the steps below. Pick 2 cards up at a time to compare values. Keep track of the number of comparisons by adding a hash mark in the space provided below.*

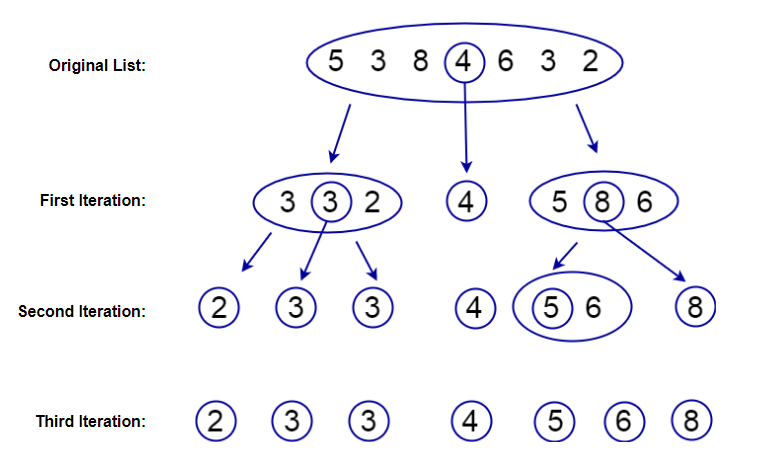
Take the first card and put it into its own sorted sublist. The sublist of 1 is sorted.

Take the second card, compare it to the first card in the sublist starting from the front. If it is smaller, insert it into the position before the card you just compared. If it is larger, compare it to the next card in the sublist until you find where it belongs.

Repeat this process with the rest of the cards.

Number of comparisons: **35 comparisons.**

**Algorithm 4**



Observe the chosen number in the original list. Compare it to the first iteration. Where are all the other numbers in the list placed relative to the chosen number? **The selected number evenly splits the group into 2 groups. It is the median. It also organized the numbers in their respective groups, larger than it on the right and smaller on the left.**

Observe the chosen numbers in the first iteration. Where are the values placed in the second iteration compared to that number. **The larger values than the first selected number are compared, selecting the then median of those, while the smaller were grouped and then a median again chosen from those.**

As a group, create a definition for this algorithm.

**While there are more than one number on either side of a selected value, select the central value, evenly splitting a list in half, and organize any larger values in a group and all smaller values in a separate group. For each of these groups, select the center point again and separate them as stated above.**

Compare your definition to the Algorithms we discussed in class. Which algorithm do you think this is? **Quick sort.**

**Sort the Cards:** *Mix your cards up and place them face down in a line on the table, sort them according to the algorithm by following the steps below. Pick 2 cards up at a time to compare values. Keep track of the number of comparisons by adding a hash mark in the space provided below.*

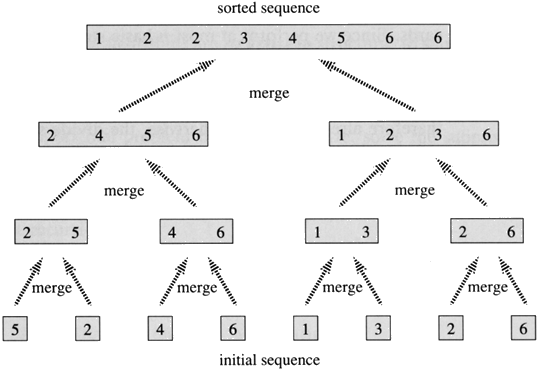
Choose the center card. Compare that card with every other card in the list one at a time.

As you compare, make two sublists, those smaller go on the left, greater goes on the right. Repeat this process with each sublist.

Number of comparisons: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Algorithm 5**

Observe the following sort and answer the questions below.



Observe the initial sequence of numbers. Each box is a sublist. How many sublists begin this algorithm? **8 sublists, the same amount as the length of the total sequence.**

Observe the next sequence up. Describe the pattern at which they are added to the next set of sublists. **2 sublists are compared to each other, and are combined into a single one, sorting them accordingly. Each list is double the length and the amount of lists is halved each time.**

As a group, create a definition for this algorithm.

**Separate every element of your sequence into their own groups. Group the separated elements and compare them, adding them together to create a single list. Repeat this until the entire sequence is a single sublist. Each iteration should half the amount of sublists.**

Compare your definition to the Algorithms we discussed in class. Which algorithm do you think this is? **Merge sort.**

**Sort the Cards:** *Mix your cards up and place them face down in a line on the table, sort them according to the algorithm by following the steps below. Pick 2 cards up at a time to compare values. Keep track of the number of comparisons by adding a hash mark in the space provided below.*

Start by separating each card so it is in it’s own sublist. Spread them apart so you can determine which lists each card belongs to.

Now that you have each element in a list by itself, you will now merge each 2 cards into a sublist of 2. As you merge the cards, check the values and place the smaller to the left and the larger to the right.

Repeat this process of merging sublists together until you have 1 sorted list.

Number of comparisons: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_