

## #7 - Tracing Sorting Algorithms - 15pts

In the table below, each row represents a 'pass' - a single iteration of the outer loop in the function. Fill in the number of comparisons and swaps that happen in each pass, and the state of the list at the **end** of that pass, for **selection sort** as implemented in class. We've filled in the first row for you.

Pass #	Comparisons	Swaps	List State
<b>Start</b>	-	-	[ 5, 4, 2, 3, 6, 7, 1 ]
<b>1</b>	<b>6</b>	<b>1</b>	[ 1, 4, 2, 3, 6, 7, 5 ]
<b>2</b>			
<b>3</b>			
<b>4</b>			
<b>5</b>			
<b>6</b>			

In the next table, each row will represent either a **split-pass** (S) or a **merge-pass** (M). Fill in the number of comparisons and copies that happened in each pass, and the state of each of the list(s) at the **end** of that pass, for **merge sort** as discussed in class. We've filled in the first row for you.

Pass #	Comparisons	Copies	List(s) State
<b>Start</b>	-	-	[ 5, 4, 2, 3, 6, 7, 1 ]
<b>S1</b>	<b>0</b>	<b>7</b>	[ 5, 4, 2 ] [ 3, 6, 7, 1 ]
<b>S2</b>			
<b>S3</b>			
<b>M1</b>			
<b>M2</b>			
<b>M3</b>			