

# Searching and Sorting Worksheet

Explore each of these common searching and sorting algorithms.

**You need to try to write out an algorithm and then draw the flow diagram for each of the searching and sorting processes.** You can draw the diagram on paper and upload it to the document.

**The Binary Search function Algorithm and Flowchart has been done for you, but you need to answer the questions**

## Binary Search

Use [this interactive](#)

Can you work out a smart way to do it in the least number of guesses?

What would be the best case?

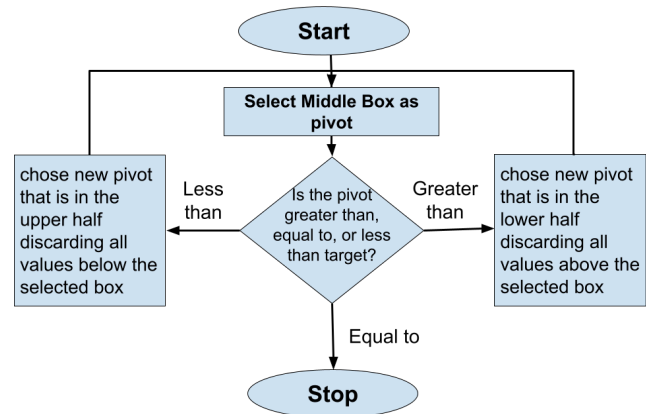
What would be the worst case?

What would be the average if you were to do it 100 times?

### ALGORITHM

1. Select the middle box
2. Determine if this middle box is greater than, equal to, or less than target
  - a. If middle box is greater than target chose new pivot that is in the lower half discarding all values above the selected box
  - b. If middle box is less than target chose new pivot that is in the upper half discarding all values below the selected box
  - c. If the value is equal to target stop

### FLOWCHART



**Following the Binary Example complete the following:**

## Linear Search

Use [this interactive](#)

How many guesses do you need for 10 boxes?

What would be the best case?

What would be the worst case?

So what do you think the average case would be for 10 boxes?

What if there were twice as many boxes? What would happen to the average case?

**Algorithm**

**Flow Chart**

## Selection Sort

Use [this interactive](#)

What do you notice about the algorithm?

How many comparisons did you have to make to be sure the data is sorted?

**Algorithm**

**Flowchart**

## Quicksort

Use [this interactive](#)

How does it compare to the selection sort?

Why do you think it is more efficient?

**Algorithm**

**Flowchart**

## Extra - Bubble Sort

Go and independently investigate bubble sort.



Algorithm	Flowchart
-----------	-----------