Common Array Algorithms

Arrays

Accessing

```
int[] myArray = {1,2,3,4,5};

for (int num : myArray) {
        System.out.println(num);
}
```

Dynamically Populating

```
int[] myArray2 = new int[5];

for (int i=0; i<myArray2.length; i++) {
        myArray2[i] = i+1;
}

// [1, 2, 3, 4, 5]
```

Printing

To print the contents of an array as a string, use the built-in `Arrays` class enclosed in a print statement:

```
System.out.println(Arrays.toString(arrayName));
```

Minimum and Maximum Values

There is no built-in method for getting the minium or maximum values of an array. There are two ways to go about it:

- Sort the array (destructive - changes the array itself)
- Store the first element in a variable and iterate through the array, comparing each value to to one in your variable (recommended)

Bubble Sort

```
static void bubbleSort(int array[]) {
    int size = array.length;
    // loop over each element of the array to access them
    for (int i = 0; i < size - 1; i++)
      // compare the elements of the array with a loop
      for (int j = 0; j < size - i - 1; j++)
        // compare two adjacent elements in the array
```

```
        if (array[j] > array[j + 1]) {
          // Swap if the elements aren't in the right order
          int temp = array[j];
          array[j] = array[j + 1];
          array[j + 1] = temp;
        }
  }
```

Storing Minimum in Variable

```
static void storeMin(int[] array) {
        int minValue = array[0];
        for (int val : array) {
                if (val < minValue) {
                        minValue = val;
                }
        }
return minValue;
}
```

2D and Multidimensional Arrays

Accessing

```
int[][] my2DArray = {{1,2,3}, {4,5,6}, {7,8,9}};

for (int i=0; i<my2DArray.length; i++) {
        for (int j=0; j<my2DArray[i].length; j++) {
                System.out.print(my2DArray[i][j] + " ");
        }
}

// 1 2 3 4 5 6 7 8 9
```

Dynamically Populating

```
int[][] my2DArray = new int[3][3];

for (int i=0; i<my2DArray.length; i++) {
        for (int j=0; j<my2DArray[i].length; j++) {
                my2DArray[i][j] = i+1;
        }
}

// {{1,2,3}, {4,5,6}, {7,8,9}}
```