Problem Solving with Code

What we're learning today:

- Common ways Arrays and Objects are used in algorithms
- How to write boolean statements as code
- How to approach problems by thinking algorithmically

Let’s look as some ways Arrays are used to solve problems

Iterating over an array

When you build a for loop like this, **array[i]** will point to one thing in the array and **i** will be it's index

```
function iterate(array) {
    for (var i=0; i < array.length; i++) {
        var currentItem = array[i];
        // use currentItem here
    }
}
```

Accumulating over an array

If you declare a variable outside of the for loop, you can mess with it in each iteration of the loop

```
function accumulate(array) {
    var count = 0;

    for (var i=0; i < array.length; i++) {
        var currentItem = array[i];
        count += currentItem;
    }

    return count;
}
```

Accumulating over an array

If you declare a variable outside of the for loop, you can mess with it in each iteration of the loop

```
function accumulate(array) {
    var count = "";

    for (var i=0; i < array.length; i++) {
        var currentItem = array[i];
        count += currentItem;
    }

    return count;
}
```

Searching for one item in an array

Use an **if** statement in your for loop to check each array item

```
function search(haystack, needle) {
    var foundIndex = -1;

    for (var i=0; i < haystack.length; i++) {
        var currentItem = haystack[i];
        if ( currentItem == needle ) {
            foundIndex = i;
        }
    }

    return foundIndex;
}
```

Divide these problems between the people at your table, discuss your solutions!

1. Given an array of people's names [“Lachlan”, “Kim”, “Moira”], write a function that logs “Hi, [name]!” for each person.
2. Given an array of a first, middle, and last name [“Adam”, “Michael”, “Szaruga”], write a function that prints the person's initials
3. Given a list of 0's and 1's, [0, 1, 0, 1, 1, 1, 0, 0], write a function that returns the index of the last 1 in the array
4. Given a list of positive numbers, [1, 5, 25, 3, 99, 20], write a function that returns the biggest number

Let’s look as some ways Objects are used to solve problems

Finding unique items in an array

Object keys are unique, whereas Array items don't have to be unique

```
function unique(array) {
    var object = {};

    for (var i=0; i < array.length; i++) {
        var currentItem = array[i];
        object[currentItem] = “blah”;
    }

    var uniqueItems = object.keys();
    return uniqueItems;
}
```

Counting unique items in an array

Use an if statement to see if the currentItem has been counted before - otherwise, just increment its count

```
function histogram(array) {
    var object = {};

    for (var i=0; i < array.length; i++) {
        var currentItem = array[i];
        if ( !object[currentItem] ) {
            object[currentItem] = 0;
        }
        object[currentItem]++;
    }

    return object;
}
```

Divide these problems between the people at your table, discuss your solutions!

1. Given an array of student's birth years, [1991, 1984, 1984, 1989], return the most common birth year
2. Given a string "This is a random string", write a function that returns a count of each character in the string
3. Given an array of olympic race results [{country: "usa", time: 233}, {country: "poland", time: 222}, ...], write a function that returns each country's best time
4. Given a list of numbers, [-1, 5, -25, -3, 99, 20], write a function that returns the count of positive numbers and negative numbers

Let’s look as some ways Arithmetic is used to solve problems

Finding multiples of a number

A number is a multiple of **n** if:
number % n == 0

```
function multiple(number, n) {

    for (var i=0; i < n; i++) {
        if ( i % number == 0 ) {
            console.log(i)
        }
    }

}
```

Finding the average of a list of numbers

average = sum/count

```
function multiple(numbers) {

    var sum = 0;
    for (var i=0; i < numbers.length; i++) {

        sum += numbers[i]

    }

    return sum/numbers.length;
}
```

Almost every problem you encounter will require you to convert plain english into Boolean logic

“i is less than 20 and greater than 0”

...

“i is less than 20” && “i is greater than 0”

...

i < 20 && i > 0

“i is either a multiple of 5 or 3”

…

“i is a multiple of 5” || “i is a multiple of 3”

…

i % 5 == 0 && i % 3 == 0

“myArray is non-empty and has an odd number of elements”

…

“myArray is non-empty” && “myArray has an odd number of elements

…

myArray.length > 0 && myArray % 2 == 1

9 times out of 10, your algorithms will use some variation of these techniques!

You can become a better problem solver if you master these techniques and recognize which work best for the problem at hand

For example, this problem....

Problem:

Write an algorithm that will add add all of the numbers between 0 and some number **n**

Solution:

```
function sum(n) {
    var sum = 0;
    for (var i=0; i < n; i++) {
        sum += i;
    }
    return sum;
}
```

... is pretty much the same as this problem!

Problem:

Write an algorithm that will add add all of the **multiples of 3** between 0 and some number **n**

Solution:

```
function sumMult3(n) {
    var sum = 0;
    for (var i=0; i < n; i++) {
        if (i % 3 == 0) {
            sum += i;
        }
    }
    return sum;
}
```

Time to tackle some exercises!

Schoology → Week 2 → Tuesday → Exercises → Problem Solving