Week 2 - Arrays 1 - Monday Recap

PushFront

I*nsert a given value* at the beginning of the given array, without using any built-in array methods.

```
function pushFront(arr, val)
{
  for (var idx = arr.length; idx > 0; idx--) {
    arr[idx] = arr[idx - 1];
  }
  arr[0] = val;
}               //  no "return arr" needed. Why?

// Arrs are objects passed by reference. A
// pointer (not a copy) of the arr is passed, so
// [] enables the method's code to cross the ptr
// and access the caller's real array

var myArray = [true, "schweet"];
pushFront(myArray, 42); // no return val needed
console.log(myArray);   // [42, true, "schweet"]
```

PopFront

Remove and return the first value of the given array, without using built-in array methods except pop().

```
function popFront(arr)
{
    if (arr.length === 0) {
        return null;
    }
    var returnVal = arr[0];

    for (var idx = 1; idx < arr.length; idx++) {
        arr[idx - 1] = arr[idx];
    }
    arr.pop();
    return returnVal;
}
```

InsertAt

Insert a given value into given array at given index, without using built-in array methods. Similar to the above, with index instead of 0.

```
function insertAt(arr, index, val)
{
  if (index < 0)          { index = 0;  }
  if (index > arr.length) { index = arr.length;}
  for (var idx = arr.length; idx > index; idx--)
  {
    arr[idx] = arr[idx - 1];
  }
  arr[index] = val;
}
```

```
// ...and now that we have the above, we can do:
function pushFront2(arr, val) {
  insertAt(arr, 0, val);
}
```

RemoveAt

Remove and return the value in a given array and index, without built-in array methods except pop(). Similar to the above, with index instead of 0.

```
function removeAt(arr, index)
{
  if (index < 0 || index >= arr.length) {
    return null;
  }
  var returnVal = arr[index];

  for (var idx = index + 1; idx < arr.length;
                            idx++) {
    arr[idx - 1] = arr[idx];
  }
  arr.pop();
  return returnVal;
}
```

```
// ...and now that we have the above, we can do:
function popFront2(arr) {
  return removeAt(arr, 0);
}
```

Week 2 - Arrays 1 - Tuesday

This week you will familiarize yourself with basic array manipulation. Here is a list of concepts to study; some or all will be used in this week's challenges.

for / while loops *array.pop() & push()* *can contain different data types*
if / else statements *arrays grow: arr.length == lastIdx-1* *arrs are objects, passed by reference (ptr)*

Reverse Array

Given a numerical array, reverse the order of the values. The reversed array should have the same length, with existing elements moved to other indices so that the order of elements is reversed.

Answer:

Remove Negatives

Implement a function removeNegatives() that accepts an array and removes any values that are less than zero. Optional: do this without two nested loops.

Answer:

Tomorrow: bread crumbs and ferris wheels