

Aaron Jones
5/16/14
Lab 9

Part 1

```
/* Write the 'array220' method in the 'Recursion' class.  
 * Given an array of ints, compute recursively if the array  
 * contains somewhere a value followed in the array by that value times 10.  
 * For example the array { 1, 3, 42, 7, 70, 0 } contains 7 followed  
 * immediately by 70 so your recursive method should return 'true'.  
 * Be sure to handle an empty array.
```

```
*/
```

```
public class Recursion  
{
```

```
    // Your recursive solution goes here -  
    // Write the 'array220' method...
```

```
    public static boolean array220(int[] nums, int index)  
    {  
        // Your code goes here...  
        int n = nums.length;  
        if(n == 0)  
        {  
            return false;  
        }  
        if(index+1 < n && nums[index]*10 == nums[index+1])  
            return true;  
        else if(index+1 < n)  
            return array220(nums, index+1);  
  
        return false;  
    }  
  
}
```

```
// Do not change this class!
```

```
public class Tester  
{
```

```
    /* Write the 'array220' method in the 'Recursion' class.  
    * Given an array of ints, compute recursively if the array  
    * contains somewhere a value followed in the array by that value times 10.  
    * For example the array { 1, 3, 42, 7, 70, 0 } contains 7 followed  
    * immediately by 70 so your recursive method should return 'true'.
```

Aaron Jones
5/16/14
Lab 9

```
* Be sure to handle an empty array.
*/
public static void main(String[] args)
{
    int[] nums1 = { 1, 3, 42, 7, 70, 0 };
    System.out.println(Recursion.array220(nums1, 0));
    int[] nums2 = { 30, 3, 42, 2, 0, 10 };
    System.out.println(Recursion.array220(nums2, 0));
    int[] nums3 = {};
    System.out.println(Recursion.array220(nums3, 0));
    int[] nums4 = { 0, -1, 42, 1, 10, 0 };
    System.out.println(Recursion.array220(nums4, 0));
    int[] nums5 = { 0, -5, 50, 1, -10, 0 };
    System.out.println(Recursion.array220(nums5, 0));
    int[] nums6 = { -1, -5, 50, 1, 0, 0 };
    System.out.println(Recursion.array220(nums6, 0));
}
}
```

Part 2

```
public class Recursion3nPlus1
{
    /* Consider the following algorithm to generate a sequence of numbers. Start
    * with an integer *n*. If *n* is even, divide by 2. If *n* is odd, multiply by
    * 3 and add 1. Repeat this process with the new value of *n*, terminating
    * when *n* = 1.
    * For example rec3nPlus1(22) prints: 22 11 34 17 52 26 13 40 20 10 5 16 8 4
    2 1
    */
    public static void rec3nPlus1(int n)
    {
        // Your code goes here...

        if(n == 0)
        {
            return;
        }

        if(n%2 == 0)
        {

```

Aaron Jones
5/16/14
Lab 9

```
        rec3nPlus1(n/2);
    }
    else
        rec3nPlus1((n * 3) + 1);
    }
}
```

// Do not change this class!

public class Tester

{

```
    public static void main ( String[] args )
    {
```

```
        /* Consider the following algorithm to generate a sequence of numbers. Start
        * with an integer *n*. If *n* is even, divide by 2. If *n* is odd, multiply by
        * 3 and add 1. Repeat this process with the new value of *n*, terminating
        * when *n* = 1.
        * For example rec3nPlus1(22) prints: 22 11 34 17 52 26 13 40 20 10 5 16 8 4
        2 1
        */
```

```
        System.out.println("3n+1 values for an input of 22:");
        Recursion3nPlus1.rec3nPlus1(22);
```

```
        System.out.println("\n\n3n+1 values for an input of 42:");
        Recursion3nPlus1.rec3nPlus1(42);
```

```
    }// end main
```

```
}// end class
```

// Do not change this class!

public class Tester

{

```
    /* Write the 'array220' method in the 'Recursion' class.
    * Given an array of ints, compute recursively if the array
    * contains somewhere a value followed in the array by that value times 10.
    * For example the array { 1, 3, 42, 7, 70, 0 } contains 7 followed
    * immediately by 70 so your recursive method should return 'true'.
    * Be sure to handle an empty array.
    */
```

```
    public static void main(String[] args)
    {
        int[] nums1 = { 1, 3, 42, 7, 70, 0 };
```

Aaron Jones

5/16/14

Lab 9

```
    System.out.println(Recursion.array220(nums1, 0));  
    int[] nums2 = { 30, 3, 42, 2, 0, 10 };  
    System.out.println(Recursion.array220(nums2, 0));  
    int[] nums3 = {};  
    System.out.println(Recursion.array220(nums3, 0));  
    int[] nums4 = { 0, -1, 42, 1, 10, 0 };  
    System.out.println(Recursion.array220(nums4, 0));  
    int[] nums5 = { 0, -5, 50, 1, -10, 0 };  
    System.out.println(Recursion.array220(nums5, 0));  
    int[] nums6 = { -1, -5, 50, 1, 0, 0 };  
    System.out.println(Recursion.array220(nums6, 0));  
    }  
}
```