

**Worksheet: Arrays Practice 2**

1. Study the following code. Method `reverseArray` is supposed to reverse the elements of an array. The code does not work as expected and, in order to debug the code, we will create a trace table.

```

1 public class ReverseArray {
2     public static void main(String[] args) {
3         int[] intArray = { 1, 3, 5, 7, 9 };
4         reverseArray(intArray);
5         System.out.println(Arrays.toString(intArray));
6     }
7     public static void swap(int[] arr, int i, int j) {
8         int tmp = arr[i];
9         arr[i] = arr[j];
10        arr[j] = tmp;
11    }
12    public static void reverseArray(int[] arr) {
13        for(int i = 0; i < arr.length; i++) {
14            int j = arr.length-i-1;
15            swap(arr, i, j);
16        }
17    }
18 }

```

- a) Complete the trace table with the values at the end each iteration (after line 15).

		arr				
i	j	0	1	2	3	4
		1	3	5	7	9
0	4	9	3	5	7	1
1	3	9	7	5	3	1
2	2	9	7	5	3	1
3	1	9	3	5	7	1
4	0	1	3	5	7	9
5	loop exited, code complete					

- b) Explain why the code does not work as intended.

*The code has the correct idea, and the array is reversed at the half-way point.*

*However, the code continues and going through the latter part of the array swaps each element pair a second time, resulting in the original array.*

- c) Write the corrected loop condition in the box, below.

```
for(int i = 0; i < arr.length/2; i++)
```

**Worksheet: Arrays Practice 2**

2. Complete the following code.

- a) Given an array, `arr`, write a statement that will declare an integer, `len`, and initialize that integer to the length of `arr`.

```
int len = arr.length;
```

- b) Write a statement that will declare an array of `String` named `arr`, that has space to store ten `String` elements.

```
String[] arr = new String[ 10 ];
```

- c) Given an array named `original`, write a statement that will declare an array of `double` named `newArr`, that is the same length as `original`.

```
double[] newArr = new double[ original.length ];
```

- d) Write a method named `makeCopy` that will take a `double` array as a parameter, make a copy of the array in reverse order, and return the new array. The original array must not be modified.

```
1 public static double[] makeCopy(double[] original)
2 {
3     int len = original.length;
4     double[] newArr = new double[ len ];
5     for(int i = 0; i < len; i++) {
6         newArr[len-i-1] = original[i];
7     }
8     return newArr;
9 }
```