University of Stirling Computing Science and Mathematics

CSCU9A1 Autumn 2016

Tutorial 5

1. Consider the following declaration of an array

```
int[] arr = { 1,2,3,4,5,6,7,8,9,0} ;
```

What is the value of `total` after each of the following fragments of code?

Hand trace some of these on the board to show what is going on.

```
int total = 0 ;
for (int i = 0 ; i < arr.length ; i++)
     {total = total + arr[i] ;}
```

45

```
int total = 0 ;
for (int i = 0 ; i < arr.length ; i=i+2)
     {total = total + arr[i] ;}
```

25

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

44

```
int total = 0 ;
for (int i = 0 ; i <= arr.length ; i=i+2)
     {total = total + arr[i] ;}
```

crashes

```
int total = 0 ;
for (int i = arr.length -1 ; i > 0 ; i=i-1)
     {total = total + arr[i] ;}
```

44

```
int total = 0 ;
for (int i = arr.length-1 ; i >= 0 ; i=i-1)
     {total = total + arr[i] ;}

     45

int total = 0 ;
for (int i = arr.length -1; i > 0 ; i=i+1)
     {total = total + arr[i] ;}

     infinite loop
```

I haven't tested the above, but I think they're correct.

2. Write a method that is passed an array, `x`, of doubles and an integer rotation amount, `n`. The method creates a new array with the items of `x` moved forward by `n` positions. Elements that are rotated off the array will be moved to the beginning. For example, suppose `x` contains the following items in sequence:

   1 2 3 4 5 6 7

   After rotating by 4, the elements in the new array will appear in this sequence:

   4 5 6 7 1 2 3

   Get the students to discuss how they would solve this problem and design the method on paper before starting to code it. The solution below is fairly straightforward. It uses a new array to hold a rotated copy of the original array. The original array is traversed, and the element at location `i` *is copied to location* `(i+shift) % array.length` *in the new array.*

```
public static double[] rotate(double[] array, int shift)
{
    double[] newArray = new double[array.length];
    for (int i = 0; i < array.length; i++)
    {
        newArray[(i+shift) % array.length] = array[i];
    }
    return newArray;
}
```