

1. What is printed when you run this code?

```
int[] nums = {3,4,7,6,9,2,11};
for(int i = nums.length-1; i>-1; i--)
{
    if(nums[i]%2==0)
        System.out.print(nums[i]+1);
}
```

3 7 5

2. What are the contents of this array when this code has run?

```
int[] nums = {3,4,7,6};
for(int i = 0; i<nums.length; i++)
{
    nums[i]=nums[i]+nums[i]%2;
}
```

{ 4, 4, 8, 6 }

Problems 3 through 5 all use the Frog class and all go together.

3. Given the Frog class to the right, write code that creates a Frog array named pond to hold 500 Frog objects and then instantiates each element.

```
Frog[] pond = new Frog[500];
for(int i = 0; i < pond.length; i++)
    pond[i] = new Frog();
```

```
class Frog
{
    private int location;
    public Frog() {location=0; }
    public void hop(int n) {location+=n; }
    public int getLocation() {return location;}
    public String toString() {String temp = "";
        for(int i = 0; i < location; i++){ temp+=".";}
        return temp+"@";
    }
}
```

4. Write code to hop each Frog in the array **pond** a random value from 10 to 50 inclusive.

```
for(int i = 0; i < pond.length; i++)
    pond[i].hop((int)(Math.random()*41)+10);
```

5. Write code to report the average location of all of the Frog objects in the **pond** array. (You will use the `getLocation()` method which returns an int value of a specific Frog object's location.) Print the average as a double.

```
int sum = 0;
for(Frog f: pond)
    sum += f.getLocation();
System.out.println("Average: " + ((double)sum/pond.length));
```

6. Write code which changes all elements of String array arr to be upper case.

```
String[] arr = //initialized with valid data;
```

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

7. Write a method `dupPresent` that accepts a `String` array named **names** and returns true if one or more duplicates are present, false if no duplicates are present. The data is not sorted.

```
public static boolean dupPresent(String[] names)
{
    for(int i = 0; i < names.length - 1; i++)
    {
        for(int j = i + 1; j < names.length; j++)
        {
            if(names[i].equals(names[j])) return true;
        }
    }
    return false;
}
```

8. Write a regular for loop to print all the elements of an int array **nums** on a single line:

```
int[] nums = //initialized with valid data;
for(int i = 0; i < nums.length; i++)
    System.out.print(nums[i] + "-");
System.out.println();
```

9. Write an enhanced for loop to add all the elements of an int array **nums** in the variable **total**.

```
int[] nums = //initialized with valid data;
int total = 0;
for(int n: nums)
    total += n;
```

10. Write code that prints all elements of the `String` array `words` which end in "ly". For example, if words contained {"coyly", "friendly", "lynx"} the code would print "coyly" and "friendly".

```
String[] words = //initialized with valid data;
```

```
for(String w: words)
{
    String lastTwoChars = w.substring(w.length() - 2);
    if(lastTwoChars.equals("ly"))
        System.out.println(w);
}
```