Unit 4 Practice Test

This test is worth 25 points:
- 12 points for the multiple-choice questions (2 points each)
- 7 points for question 7, and 6 points for question 8
You may specify up to two answers for each multiple-choice question. You will get one point for any problem in which your first choice is wrong but your second choice is correct.

1st  2nd

_____ 1. What is the output of the following Java code fragment?

```java
int[] values = {3, 5, 1, 10, 7};
System.out.println(values[2] + " " + values.length);
```

A.  5  4
B.  5  5
C.  1  4
D.  1  5
E.  none of these

_____ 2. What is the output of the following Java code fragment?

```java
int[] f = {2, 4, 6, 8, 10, 12};
int[] g = new int[6];
for (int i = 0; i < 6; i++) {
g[5 - i] = f[i];
}
System.out.println(f[3] + " " + g[3]);
```

A.  8  6
B.  6  8
C.  8  8
D.  6  6
E.  none of these

_____ 3. What is the output of the following Java code fragment?

```java
String s = "With Crimson in triumph flashing";
String[] t = s.split("i");
System.out.println(t[2] + "&" + t[t.length - 1]);
```

A.  in&flashing
B.  Crimson&triumph
C.  mson &ng
D.  th Cr&ng
E.  th Cr&flash
F.  none of the above
4. Consider the following Java program:

```java
import java.util.*;

public class Problem4 {
    public static void main(String[] args) {
        int[] x = {5, 6, 7};
        int y = 4;
        mystery(x, y);
        System.out.println(Arrays.toString(x) + " " + y);
    }

    public static void mystery(int[] z, int y) {
        for (int i = 0; i < z.length; i++) {
            z[i] += y;
        }
        y *= 2;
        System.out.println(Arrays.toString(z));
    }
}
```

What does it output?

A.  
   [9, 10, 11] 
   [5, 6, 7] 8

B.  
   [9, 10, 11] 
   [5, 6, 7] 4

C.  
   [9, 10, 11] 
   [9, 10, 11] 8

D.  
   [9, 10, 11] 
   [9, 10, 11] 4

E.  
   [13, 14, 15] 
   [5, 6, 7] 8

F.  none of these

5. What is the output of the following Java code fragment?

```java
int[][] h = {{1, 3, 5, 7},
            {2, 4, 6, 8},
            {9, 11, 13, 15}};
System.out.println(h[2][1] + " " + h.length);
```

A.  6 3
B.  11 3
C.  4 4
D.  11 4
E.  6 12
F.  none of these
6. Consider the following recursive method:

```java
public static void mystery6(String s) {
    if (s.length() <= 1) {
        System.out.println("$");
        return;
    }
    mystery6(s.substring(0, s.length() - 1));
    System.out.print(s.charAt(s.length() - 1));
}
```

What is output by the method call `mystery6("Cambridge")`?

A. $ambridge
B. $Cambridge
C. egdirbmac$
D. egdirbma$
E. none of the above
7. Consider a text file called fight.txt that contains the following 10 lines:
   
   Fight fiercely, Harvard!
   Fight, fight, fight!
   Demonstrate to them our skill.
   Albeit they possess the might,
   Nonetheless we have the will.
   How we shall celebrate our victory?
   We shall invite the whole team
   Up for tea! How jolly!
   Hurl that spheroid down the field
   And fight! Fight! Fight!
   
   Complete the following template to create a program that counts the number of words in this file. Note that we are using the line-based approach to file processing that we focused on in lecture.

   ```java
   import java.util.*;
   import java.io.*;

   public class FierceFileProcessor {
       public static void main(String[] args)
           throws ____________________________ {

           Scanner input = new Scanner(______________________);

           // Put any lines that are needed before the loop here.

           while (______________________________) {
               String s = input.nextLine();

               // Put the rest of the loop body below.

           }

           System.out.println("The song has " + ________________
               + "; words.");
       }
   } // (song lyrics by Tom Lehrer '47)
   ```
Write a static method `minGap` that takes an array of integers as a parameter and that returns the minimum gap between adjacent values in the array. The gap between two adjacent values in an array is defined as the second value minus the first value. For example, suppose that you have the following array:

```java
int[] list = {1, 3, 7, 2, 12};
```

The first gap is 2 (3 – 1), the second gap is 4 (7 – 3), the third gap is -5 (2 – 7), and the fourth gap is 10 (12 – 2). Thus, the call:

```java
minGap(list)
```

should return -5, because that is the smallest gap in the array. Note that negative gaps are treated as smaller than non-negative gaps.

If the method is passed an array with fewer than 2 elements, it should return 0.