CS159 – Advanced Programming **James Madison University**
Sample Examination 1 **Spring 2024**

This work complies with the JMU Honor Code.

Name:_____ Signature:_____

Instructions. Answer all of the following questions. This is a "closed book" examination and you must work entirely on your own. You may not use a computing or communications device of any kind. Your answers need not conform to the course style guide. All questions that involve code or are about code use the Java programming language.

1. (10 points) Choose the best answer to each of the following:

(1) ____ In Java, `<` is

a. A relational operator
b. A unary operator
c. A distributed operator
d. All of the above
e. None of the above

(2) ____ In Java, `&&`

a. Is a logical operator
b. Is a binary operator
c. Has boolean operands
d. All of the above
e. None of the above

(3) ____ In Java, the `main()` method of the "main class" must have what return type?

a. double
b. int
c. String
d. String[]
e. `void`

(4) ____ In Java, the `main()` method of the "main class" must have a parameter of what type?

a. double
b. int
c. String
d. String[]
e. void

(5) ____ In Java, what kind of **statement** is `int i;`?

a. Assignment statement
b. Declaration statement
c. Typecast statement
d. All of the above
e. None of the above

2: Given the following variable declarations and initialization:

```
boolean isCold = false;
char indicator = 'h';
double temperature = 30.99;
String[] weathers = {"cloud", "rain", "sunny"};
String city = "Harrisonburg";
int nums = 100;
```

indicate whether each of the following assignments is valid (`V`) or invalid (`I`).

2.1 `weathers[0] = "wind";` _____

2.2 `weathers[3] = "snow";`_____

2.3 `city.length() = nums;`_____

2.4 `indicator = city.substring(0, 1);`_____

2.5 `temperature = 90;` _____

2.6 `nums += 13;` _____

2.7 `indicator = city.charAt(2);` _____

2.8 `isCold = isCold + 1;` _____

2.9 `city = city.toUpperCase();`_____

2.10 `indicator = city.indexOf("son");`_____

3. For each Java expression in the table below, write the resulting value in the Result Value column and the type of the value in the Type column. If the expression would result in an error, write "error" in both columns.

Expression	Results	Type
`2.5 + 10 + "Hot"`		
`0.1 + 0.1 + 0.1 == 0.3`		
`2 + 5 / 2`		
`10 % 3 - 3 / 2.0`		
`1 < 2 && true`		
`10 + "JMU" + 1`		
`10 || 3 < 4`		
`11.2 > 1 == 3`		
`1 < 3 - 10`		

4. Write a single java statement to accomplish each of the following:

4.1: Declare a variable `str` that is a String and initialize it to "`Lucky`"

4.2 Declare a variable `arr` that is an array of String with 10 elements.

4.3 Initialize the second entry of the `arr` as value of `str`.

4.4 Write a statement to converts `str` to lower case letters

4.5 Declare a String variable named `sub` and assign it the value of `str` excluding the last two characters using the `substring` method.

4.6 Declare a variable `len` that is `int` and assign it with the length of the `str`.

5. You are given the following code.

```
public static String whatToWear(int rain, int temp) {
     String wearing = "shirt";
     if (rain > 25) {
           wearing = "jacket";
     } else {
           if (temp == 30) {
               wearing = "coat";
            }
     }
    return wearing;
}
```

Determine the value returned when the method is invoked as follows:

5.1 `whatToWear(24, 30)` _____

5.2 `whatToWear(12, 70)`_____

5.3 `whatToWear(79, 30)`_____

6. What is printed when the code segment below is executed?

6.1 _____

```
String[] names = {"pongo", "nina", "sammie"};

for (int i = 0; i <= 2; i++) {
  System.out.print(i);
  System.out.print(":");
  System.out.println(names[i]);
}
```

6.2_____

```
String[] names = {"pongo", "nina", "sammie"};

int i = 0;
while (i < names.length) {
      System.out.print(i);
      System.out.print(":");
      System.out.println(names[i]);
      i += 2;
}
```

7 Write a method that will return the count of the integers divisible by three in the provided array.

Example:

```
Input: {1, 2, 3, 4, 5, 6}
Return: 2
public class Question7 {
     public static int countThreeDivisible (int[] numArray) {









  }
}
```

8. Given the `countThreeDivisible` method above(question 7), complete the test method for it below.

Hints: You need to write done at least four statements, one statement to build the input array, two assignment statements to set up actual and expected values, and then an appropriate assert statement.

```
import static org.junit.jupiter.api.Assertions.*;
import org.junit.jupiter.api.Test;

public class Question7Test {

     @Test
     public void testCountThreeDivisible() {









     }
}
```

9. Write a method `surfaceArea` to return the surface area of a cylinder. The surface area of a cylinder is

$$2\pi r^2 + 2\pi rh,$$

where `r` is a radius and `h` is the height of the cylinder. Use the `PI` constant from the `Math` Class for π: `Math.PI`.

```
public class Question9 {
     public static double surfaceArea(double radius, double height) {









  }
}
```

10. Given the `surfaceArea` method above(question 10), complete the test method for it below.

Hints: You need to write done at least three statements, two assignment statements to set up actual and expected values and then an appropriate assert statement. Returned values within 0.000001 of the expected value should pass the test.

```
import static org.junit.jupiter.api.Assertions.*;
import org.junit.jupiter.api.Test;

public class Question9Test {
     @Test
     public void testSurfaceArea () {









     }
```

```
}
```

11.Write a method that will return a copy of the provided array, maintaining its original length. In the copied array, each entry should mirror the corresponding entry from the original array if it starts with the character `start` and ends with the character `end` at the same index. If an entry in the original array doesn't meet this condition, the corresponding position in the copied array should be with the String "`pongo`".

(You may assume that the array and all array entries are properly initialized and each entry is with a length of 1 at least.)

For example:

```
Input: {"clouds", "is", "c", "jmu", "cows"}, 'c', 's'
Return: {"clouds", "pongo", "pongo", "pongo", "cows"}

Input: {"I", "insert", "bowling", "impact", "invent"}, 'i', 't'
Return: {"pongo", "insert", " pongo", "impact", "invent"}
```

Use the following declarations:

```
public static String[] specialEntries(String[] words, char start,
char end)
{




}
```