Circle the correct answer for each of the fifteen multiple choice questions. Each multiple choice question is worth 4 points.

1) If you want to output the text "hi there", including the quote marks, which of the following could do that?
   a) System.out.println("hi there");
   b) System.out.println(""hi there"");
   c) System.out.println("\"hi there\"\");
   d) System.out.println(""hi there\"\"");
   e) none, you cannot output a quote mark since it marks the start and end of the String to be output.

   The correct answer is (d). The backslash character is used to escape the usual meaning of the quote character so that it does not terminate the string but is a character in the string itself. Choice (a) does not produce the quote marks, choices (b) and (c) produce compile errors.

2) What value will z have after the statement: float z = 5 / 10;
   a) z will equal 0.0
   b) z will equal 0.5
   c) z will equal 5.0
   d) z will equal 0.05
   e) none of the above, a compile-time error arises because z is a float and 5 / 10 is an int

   The correct answer is (a) since integer division truncates any fractional part, resulting in the integer quotient of zero. When this is assigned to a float variable, it still has the value zero.

3) What value will z have after the statement: int z = 50 / 10.00;
   a) 5
   b) 5.0
   c) 50
   d) 10
   e) none of the above, a compile-time error arises because z is an int and 50 / 10.00 is not

   The correct answer is (e) since the result of dividing an integer by a double is a double, and the compiler flags that as an error when the double value is assigned to an integer.

4) Assume b is 5 and c is 10 and both are int’s. What will be the value of a after the statement:
   int a = b * (-c + 2) / 2;
   a) 30
   b) -30
   c) 20
   d) -20
   e) -6

   The correct answer is (d).

5) What does System.out.println(x+y) output if x and y are int values with x=10 and y=5?
   a) 15
   b) 105
   c) 10.5
   d) x+y
   e) An error since neither x nor y is a String

   The correct answer is (a) since the expression x+y is evaluated to the result of 15 before being passed to println where it is converted to the String representation.
6) What does `System.out.println(""+x+y)` output if `x` and `y` are int values with `x=10` and `y=5`?
   a) 15
   b) 105
   c) 105
   d) x+y
   e) An error since neither `x` nor `y` is a String

   The correct answer is (b) since the initial operand in the expression is the empty String " ", forcing the next integer operand to be converted to a String and concatenated, and then the third operand is converted to a String and concatenated. Note this happens since the plus operator associates from left to right.

7) Which of the following would return the last character of the non-empty String `s`?
   a) `s.charAt(0);`
   b) `s.charAt(last);`
   c) `s.charAt(length(s));`
   d) `s.charAt(s.length()-1);`
   e) `s.charAt(s.length());`

   The correct answer is (d) since the first character of a String is at index 0 and the last character of a String is at index one less than the length of the String.

The next four questions refer to the following recursive factorial method:
```java
public int factorial(int x) {
    if (x > 1) return x * factorial(x - 1);
    else return 1;
}
```

8) What is returned if `factorial(3)` is called?
   a) 0
   b) 1
   c) 3
   d) 6
   e) 9

   The correct answer is (d).

9) What is returned if `factorial(0)` is called?
   a) 0
   b) 1
   c) 2
   d) nothing, `factorial(0)` causes infinite recursion
   e) nothing, `factorial(0)` produces a run-time error

   The correct answer is (b).

10) How many times is the `factorial` method invoked if originally called with `factorial(5)`?
    Include the original method call in your counting.
    a) 1
    b) 4
    c) 5
    d) 6
    e) 7

    The correct answer is (c).
11) What condition defines the base case for this method?
   a) $x$ is greater than 1
   b) $x$ is equal to 1
   c) $x$ is equal to 0
   d) $x$ is less than or equal to 0
   e) $x$ is less than or equal to 1

   The correct answer is (e) which is the negation of the condition for the recursion.

12) Consider the following outline of a nested if-else structure that has more if clauses than else clauses.

   Which of the statements below is true regarding this structure?
   
   ```java
   if (condition1)
     if (condition2)
       statement1;
     else statement2;
   
   a) syntactically it is invalid to have more if clauses than else clauses
   b) statement2 will only execute if condition1 is false and condition2 is false
   c) statement2 will only execute if condition1 is true and condition2 is false
   d) statement2 will only execute if condition1 is false, it does not matter what condition2 is
   e) statement2 will never execute
   
   The correct answer is (c) since the else construct belongs to the closest if, regardless of indentation.

13) The statement
   ```java
   if (x < 0) y = x; else y = 0;
   ```
   can be rewritten using a conditional operator as
   a) $y = (x < 0) ? x : 0$;
   b) $x = (x < 0) ? y : 0$;
   c) $(x < 0) ? y = x : y = 0$;
   d) $y = (x < 0)$;
   e) $y = if (x < 0) x : 0$;

   The correct answer is (a). Clearly the assignment is to the variable $y$, the condition is $(x < 0)$, and the alternative values assigned to $y$ are either $x$ or 0.

14) Given that $s$ is a String, what does the following loop do?
   ```java
   for(int j = s.length(); j > 0; j --)
     System.out.print(s.charAt(j-1));
   ```
   a) it prints $s$ out backwards
   b) it prints $s$ out forwards
   c) it prints $s$ out backwards after skipping the last character
   d) it prints $s$ out backwards but does not print the 0th character
   e) it yields a run-time error because there is no character at $s.charAt(j-1)$ for $j=0$

   The correct answer is (a). Note that even though the variable $j$ goes from the string length to 1, the index used to get characters from the string is one less, i.e., from the length minus one to zero, which is exactly the characters of the string in reverse order.

15) A class constructor typically is used to define
   a) how an object is initialized
   b) how an object is interfaced
   c) the number of instance data in the class
   d) the number of methods in the class
   e) if the instance data are accessible outside of the object directly

   The correct answer is (a). Initialization is the main purpose of a constructor.
16) [8 points] Complete the justDigits method of the following class. You may determine whether a character ch is a digit by calling the static method Character.isDigit(ch) from the Java API. This method returns true if ch is a digit character and false otherwise.

```java
public class PhoneNumber {
    public static void main(String args[]) {
        // Reduce a phone number to just the digits
        System.out.println("The number is: " + justDigits(args[0]));
    }

    // Return a String that contains just the digits of the given string.
    // For example, justDigits("(541) 555-1212") would return "5415551212"
    // and justDigits("+1.800.800.1000") would return "18008001000"
    public static String justDigits(String s) {
        String result = "";
        for (int i = 0; i < s.length(); ++i) {
            if (Character.isDigit(s.charAt(i)))
                result += s.charAt(i);
        }
        return result;
    }
}
```

17) [8 points] Fill in the missing entries in the following table where the numbers in each row are equivalent, in different bases as labeled.

<table>
<thead>
<tr>
<th>Decimal</th>
<th>Binary</th>
<th>Octal</th>
<th>Hexadecimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>102</td>
<td>01100110</td>
<td>146</td>
<td>66</td>
</tr>
<tr>
<td>5</td>
<td>00000101</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>172</td>
<td>10101100</td>
<td>254</td>
<td>AC</td>
</tr>
<tr>
<td>205</td>
<td>11001101</td>
<td>315</td>
<td>CD</td>
</tr>
</tbody>
</table>

18) [8 points] Combine the following five assignment statements into a single statement using addition and subtraction, prefix and postfix increment and decrement operators, and parentheses for grouping as necessary. Your single statement must have the same effect on the variables x, y, z as the five statements. Assume all variables are int variables.

```
x = y - 3;
y = y + 1;
z = z - 1;
x = x - z;
x = x + 1;
```

```
x = (y++ - 3) - --z + 1  or simplified to  x = y++ - 2 - --z
```
19) [8 points] Suppose you have a staircase and can climb the stairs either one or two stairs at a time. For example, if there are three stairs, there are three possible ways you could climb them: 1-1-1 or 1-2 or 2-1, where the digit represents whether you are climbing one or two stairs on that step. If there are four stairs, you could climb them in five possible ways: 1-1-1-1 or 1-1-2 or 1-2-1 or 2-1-1 or 2-2.

Let waysToClimb(n) represent the number of different ways to climb a staircase of n stairs, taking one or two stairs at a time. Give a recursive definition of waysToClimb:

Base case(s):

If n is 1, waysToClimb(n) is 1
If n is 2, waysToClimb(n) is 2

Recursive case:

If n is greater than 2, then waysToClimb(n) is: waysToClimb(n-1) + waysToClimb(n-2)

You can see this by thinking about the first step. If you take one stair, there are n-1 left and so waysToClimb(n-1) ways to finish climbing the stairs. If you take two stairs on the first step, there are n-2 stairs left and so waysToClimb(n-2) ways to finish climbing the stairs. This covers all the possibilities, so the sum of these two quantities is the total ways to climb n stairs.

20) [8 points] The definitions for classes Circle and Rect have been started below. These classes correspond to the concepts of circles and rectangles and they both implement the Java interface named Shape defined on the following page. Fill in the missing code, including all necessary constructors and methods, so that the classes Circle and Rect will compile and conform to the behavior defined by Shape, and will produce the results of the test driver shown on the following page. (You can detach the following page, but code your answer below.)

```java
public class Circle implements Shape {
    private int x, y;    // Coordinates of center
    private int radius;  // length of radius

    public Circle(int r) { x = 0; y = 0; radius = r; }

    public void moveBy(int x_offset, int y_offset) { x += x_offset; y += y_offset; }

    public void resizeBy(int factor) { radius += factor; }

    public float area() { return (float)(Math.PI * radius * radius); }

    public String toString() {
        return "Circle with center (" + x + "," + y + ") and radius " + radius;
    }
}
```
public class Rect implements Shape {
    private int x, y;           // Coordinates of center
    private int width, height;  // dimensions

    public Rect(int w, int h) { x = 0; y = 0; width = w; height = h; }

    public float area() { return width * height; }

    public void moveBy(int x_offset, int y_offset) { x += x_offset; y += y_offset; }

    public void resizeBy(int amount) { width += amount; height += amount; }

    public String toString() {
        return width + "x" + height + " Rectangle with center (" + x + "," + y + ")";
    }
}

Note that we must provide constructors for each of the classes, and we can look at the driver code to see what arguments must be given to constructors. We must also provide each method listed in the interface that is not already provided in each class.
This is the definition of the interface Shape:

```java
public interface Shape {
    // Move the position of the Shape by the given amounts
    public void moveBy(int x_offset, int y_offset);

    // Change the dimension of the Shape by the amount
    public void resizeBy(int amount);

    // Compute the area of the Shape
    public float area();

    // Realize the Shape as a String
    public String toString();
}
```

This is the definition of the driver program to test the Circle and Rect classes:

```java
public class ShapeTest {
    public static void main(String args[]) {
        Shape c = new Circle(5);
        Shape r = new Rect(6, 7);

        System.out.println("c is a " + c);
        System.out.println("r is a " + r);

        c.moveBy(2,1);
        r.moveBy(6,7);
        r.moveBy(-2,-2);
        c.resizeBy(4);
        r.resizeBy(-3);
        System.out.println("now c is a " + c);
        System.out.println("now r is a " + r);
        System.out.println("The area of c is " + c.area());
        System.out.println("The area of r is " + r.area());
    }
}
```

This is the output from running ShapeTest:

```
c is a Circle with center (0,0) and radius 5
r is a 6x7 Rectangle with center (0,0)
now c is a Circle with center (2,1) and radius 9
The area of c is 254.46901
now r is a 3x4 Rectangle with center (4,5)
The area of r is 12.0
```