

- j.  $z$  is nonnegative.
- k.  $x$  is even.
- l.  $x$  is closer in value to  $y$  than  $z$  is.

2. Given the variable declarations

```
int x = 4;
int y = -3;
int z = 4;
```

what are the results of the following relational expressions?

- a.  $x == 4$
- b.  $x == y$
- c.  $x == z$
- d.  $y == z$
- e.  $x + y > 0$
- f.  $x - z != 0$
- g.  $y * y <= z$
- h.  $y / y == 1$
- i.  $x * (y + 2) > y - (y + z) * 2$

3. Which of the following if statement headers uses the correct syntax?

- a. if  $x = 10$  then {
- b. if [ $x == 10$ ] {
- c. if ( $x ==> y$ ) {
- d. if ( $x$  equals 42) {
- e. if ( $x == y$ ) {

4. The following program contains 7 mistakes! What are they?

```
1 public class Oops4 {
2     public static void main(String[] args) {
3         int a = 7, b = 42;
4         minimum(a, b);
5         if {smaller = a} {
6             System.out.println("a is the smallest!");
7         }
8     }
9
10    public static void minimum(int a, int b) {
11        if (a < b) {
12            int smaller = a;
13        } else (a ==> b) {
14            int smaller = b;
15        }
16        return int smaller;
17    }
18 }
```

5. Consider the following method:

```
public static void ifElseMystery1(int x, int y) {  
    int z = 4;  
    if (z <= x) {  
        z = x + 1;  
    } else {  
        z = z + 9;  
    }  
    if (z <= y) {  
        y++;  
    }  
    System.out.println(z + " " + y);  
}
```

What output is produced for each of the following calls?

- a. `ifElseMystery1(3, 20);`
- b. `ifElseMystery1(4, 5);`
- c. `ifElseMystery1(5, 5);`
- d. `ifElseMystery1(6, 10);`

6. Consider the following method:

```
public static void ifElseMystery2(int a, int b) {  
    if (a * 2 < b) {  
        a = a * 3;  
    } else if (a > b) {  
        b = b + 3;  
    }  
    if (b < a) {  
        b++;  
    } else {  
        a--;  
    }  
    System.out.println(a + " " + b);  
}
```

What output is produced for each of the following calls?

- a. `ifElseMystery2(10, 2);`
- b. `ifElseMystery2(3, 8);`
- c. `ifElseMystery2(4, 4);`
- d. `ifElseMystery2(10, 30);`

7. Write Java code to read an integer from the user, then print even if that number is an even number or odd otherwise. You may assume that the user types a valid integer.



```

if (times == 2) {
    System.out.print("And how much are you contributing? ");
    int donation = console.nextInt();
    sum = sum + 2 * donation;
    count2++;
    total = total + donation;
}

```

Rewrite it so that it has a better structure and avoids redundancy. To simplify things, you may assume that the user always types 1 or 2. (How would the code need to be modified to handle any number that the user might type?)

12. The following code is poorly structured:

```

Scanner console = new Scanner(System.in);
System.out.print("How much will John be spending? ");
double amount = console.nextDouble();
System.out.println();
int numBills1 = (int) (amount / 20.0);
if (numBills1 * 20.0 < amount) {
    numBills1++;
}
System.out.print("How much will Jane be spending? ");
amount = console.nextDouble();
System.out.println();
int numBills2 = (int) (amount / 20.0);
if (numBills2 * 20.0 < amount) {
    numBills2++;
}
System.out.println("John needs " + numBills1 + " bills");
System.out.println("Jane needs " + numBills2 + " bills");

```

Rewrite it so that it has a better structure and avoids redundancy. You may wish to introduce a method to help capture redundant code.

13. Write a piece of code that reads a shorthand text description of a color and prints the longer equivalent. Acceptable color names are B for Blue, G for Green, and R for Red. If the user types something other than B, G, or R, the program should print an error message. Make your program case-insensitive so that the user can type an uppercase or lowercase letter. Here are some example executions:

```

What color do you want? B
You have chosen Blue.

```

```

What color do you want? g
You have chosen Green.

```

```

What color do you want? Bork
Unknown color: Bork

```



14. Write a piece of code that reads a shorthand text description of a playing card and prints the longhand equivalent. The shorthand description is the card's rank (2 through 10, J, Q, K, or A) followed by its suit (C, D, H, or S). You should expand the shorthand into the form "<Rank> of <Suit>". You may assume that the user types valid input. Here are two sample executions:

```
Enter a card: 9 S
Nine of Spades
```

```
Enter a card: K C
King of Clubs
```

#### Section 4.2: Cumulative Algorithms

15. What is wrong with the following code, which attempts to add all numbers from 1 to a given maximum? Describe how to fix the code.

```
public static int sumTo(int n) {
    for (int i = 1; i <= n; i++) {
        int sum = 0;
        sum += i;
    }
    return sum;
}
```

16. What is wrong with the following code, which attempts to return the number of factors of a given integer  $n$ ? Describe how to fix the code.

```
public static int countFactors(int n) {
    for (int i = 1; i <= n; i++) {
        if (n % i == 0) { // factor
            return i;
        }
    }
}
```

17. Write code to produce a cumulative product by multiplying together many numbers that are read from the console.
18. The following expression should equal 6.8, but in Java it does not. Why not?

```
0.2 + 1.2 + 2.2 + 3.2
```

19. The following code was intended to print a message, but it actually produces no output. Describe how to fix the code to print the expected message.

```
double gpa = 3.2;
if (gpa * 3 == 9.6) {
    System.out.println("You earned enough credits.");
}
```



**Section 4.3: Text Processing**

20. What output is produced by the following program?

```

1 public class CharMystery {
2     public static void printRange(char startLetter, char endLetter) {
3         for (char letter = startLetter; letter <= endLetter; letter++) {
4             System.out.print(letter);
5         }
6         System.out.println();
7     }
8
9     public static void main(String[] args) {
10        printRange('e', 'g');
11        printRange('n', 's');
12        printRange('z', 'a');
13        printRange('q', 'r');
14    }
15 }

```

21. Write an `if` statement that tests to see whether a `String` begins with a capital letter.

22. What is wrong with the following code, which attempts to count the number occurrences of the letter 'e' in a `String`, case-insensitively?

```

int count = 0;
for (int i = 0; i < s.length(); i++) {
    if (s.charAt(i).toLowerCase() == 'e') {
        count++;
    }
}

```

23. Consider a `String` stored in a variable called `name` that stores a person's first and last name (e.g., "Marla Singer").

Write the expression that would produce the last name followed by the first initial (e.g., "Singer, M.").

24. Write code to examine a `String` and determine how many of its letters come from the second half of the alphabet (that is, have values of 'n' or subsequent letters). Compare case-insensitively, such that values of 'N' through 'Z' also count. Assume that every character in the `String` is a letter.

**Section 4.4: Methods with Conditional Execution**

25. Consider a method `printTriangleType` that accepts three integer arguments representing the lengths of the sides of a triangle and prints the type of triangle that these sides form. The three types are equilateral, isosceles, and scalene. An equilateral triangle has three sides of the same length, an isosceles triangle has two sides that are the same length, and a scalene triangle has three sides of different lengths.

However, certain integer values (or combinations of values) would be illegal and could not represent the sides of an actual triangle. What are these values? How would you describe the precondition(s) of the `printTriangleType` method?



26. Consider a method `getGrade` that accepts an integer representing a student's grade percentage in a course and returns that student's numerical course grade. The grade can be between 0.0 (failing) and 4.0 (perfect). What are the preconditions of such a method?
27. The following method attempts to return the median (middle) of three integer values, but it contains logic errors. In what cases does the method return an incorrect result? How can the code be fixed?

```
public static int medianOf3(int n1, int n2, int n3) {
    if (n1 < n2) {
        if (n2 < n3) {
            return n2;
        } else {
            return n3;
        }
    } else {
        if (n1 < n3) {
            return n1;
        } else {
            return n3;
        }
    }
}
```

28. One of the exercises in Chapter 3 asked you to write a method that would find the roots of a quadratic equation of the form  $ax^2 + bx + c = 0$ . The quadratic method was passed  $a$ ,  $b$ , and  $c$  and then applied the following quadratic formula:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Under what conditions would this formula fail? Modify the quadratic method so that it will reject invalid values of  $a$ ,  $b$ , or  $c$  by throwing an exception. (If you did not complete the exercise in the previous chapter, just write the method's header and the exception-throwing code.)

29. Consider the following Java method, which is written incorrectly:

```
// This method should return how many of its three
// arguments are odd numbers.
public static void printNumOdd(int n1, int n2, int n3) {
    int count = 0;
    if (n1 % 2 != 0) {
        count++;
    } else if (n2 % 2 != 0) {
        count++;
    }
```



14. Modify your `pow` method from Exercise 5 to make a new method called `pow2` that uses the type `double` for the first parameter and that works correctly for negative numbers. For example, the call `pow2(-4.0, 3)` should return `-4.0 * -4.0 * -4.0`, or `-64.0`, and the call `pow2(4.0, -2)` should return `1 / 16`, or `0.0625`.
15. Write a method called `getGrade` that accepts an integer representing a student's grade in a course and returns that student's numerical course grade. The grade can be between 0.0 (failing) and 4.0 (perfect). Assume that scores are in the range of 0 to 100 and that grades are based on the following scale:

Score	Grade
< 60	0.0
60–62	0.7
63	0.8
64	0.9
65	1.0
...	
92	3.7
93	3.8
94	3.9
>= 95	4.0

For an added challenge, make your method throw an `IllegalArgumentException` if the user passes a grade lower than 0 or higher than 100.

16. Write a method called `printPalindrome` that accepts a `Scanner` for the console as a parameter, prompts the user to enter one or more words, and prints whether the entered `String` is a palindrome (i.e., reads the same forward as it does backward, like "abba" or "racecar").
- For an added challenge, make the code case-insensitive, so that words like "Abba" and "Madam" will be considered palindromes.
17. Write a method called `stutter` that accepts a string parameter and returns that string with its characters repeated twice. For example, `stutter("Hello!")` returns `HHeeellllloo!!`.
18. Write a method called `wordCount` that accepts a `String` as its parameter and returns the number of words in the `String`. A word is a sequence of one or more nonspace characters (any character other than ' '). For example, the call `wordCount("hello")` should return 1, the call `wordCount("how are you?")` should return 3, the call `wordCount("this string has wide spaces")` should return 5, and the call `wordCount(" ")` should return 0.