Pay special attention to this line from the program:

```
result = i != 0 ? 100 / i : 0;
```

Here, **result** is assigned the outcome of the division of 100 by **i**. However, this division takes place only if **i** is not zero. When **i** is zero, a placeholder value of zero is assigned to **result**.

You don't actually have to assign the value produced by the **?** to some variable. For example, you could use the value as an argument in a call to a method. Or, if the expressions are all of type **boolean**, the **?** can be used as the conditional expression in a loop or **if** statement. For example, here is the preceding program rewritten a bit more efficiently. It produces the same output as before.

```
// Prevent a division by zero using the ?.
class NoZeroDiv2 {
  public static void main(String args[]) {

    for(int i = -5; i < 6; i++)
      if(i != 0 ? true : false)
        System.out.println("100 / " + i +
                           " is " + 100 / i);
  }
}
```

Notice the **if** statement. If **i** is zero, then the outcome of the **if** is false, the division by zero is prevented, and no result is displayed. Otherwise the division takes place.

Module 5 Mastery Check

1. Show two ways to declare a one-dimensional array of 12 **double**s.
2. Show how to initialize a one-dimensional array of integers to the values 1 through 5.
3. Write a program that uses an array to find the average of 10 **double** values. Use any 10 values you like.
4. Change the sort in Project 5-1 so that it sorts an array of strings. Demonstrate that it works.
5. What is the difference between the **String** methods **indexOf( )** and **lastIndexOf( )**?
6. Since all strings are objects of type **String**, show how you can call the **length( )** and **charAt( )** methods on this string literal: "I like Java".

7. Expanding on the **Encode** cipher class, modify it so that it uses an eight-character string as the key.

8. Can the bitwise operators be applied to the **double** type?

9. Show how this sequence can be rewritten using the **?** operator.

```
if(x < 0) y = 10;
else y = 20;
```

10. In the following fragment, is the **&** a bitwise or logical operator? Why?

```
boolean a, b;
// ...
if(a & b) ...
```

11. Is it an error to overrun the end of an array? Is it an error to index an array with a negative value?

12. What is the unsigned right-shift operator?

13. Rewrite the **MinMax** class shown earlier in this chapter so that it uses a for-each style **for** loop.

14. Can the **for** loops that perform sorting in the **Bubble** class shown in Project 5-1 be converted into for-each style loops? If not, why not?