

1. Which of the following pairs of declarations will cause an error message?

- I double x = 14.7;  
int y = x;
- II double x = 14.7;  
int y = (int) x;
- III int x = 14;  
double y = x;

- (A) None
- (B) I only
- (C) II only
- (D) III only
- (E) I and III only

2. What output will be produced by

```
System.out.print("\\* This is not\n a comment *\\");
```

- (A) \* This is not a comment \*
- (B) \\\* This is not a comment \*\\
- (C) \* This is not  
a comment \*
- (D) \\\* This is not  
a comment \*\\
- (E) \\\* This is not  
a comment \*\\

3. Refer to the following code fragment:

```
double answer = 13/5;  
System.out.println("13/5 = " + answer);
```

The output is

13/5 = 2.0

The programmer intends the output to be

13/5 = 2.6

Which of the following replacements for the first line of code will *not* fix the problem?

- (A) double answer = (double) 13/5;
- (B) double answer = 13/(double) 5;
- (C) double answer = 13.0/5;
- (D) double answer = 13/5.0;
- (E) double answer = (double) (13/5);

4. What value is stored in result if

```
int result = 13 - 3 * 6 / 4 % 3;
```

- (A) -5
- (B) 0
- (C) 13
- (D) -1
- (E) 12

6. Let  $x$  be a variable of type `double` that is positive. A program contains the boolean expression `(Math.pow(x, 0.5) == Math.sqrt(x))`. Which of the following is the most likely reason why this expression can have the value `false`?

- (A)  $x^{1/2}$  is not mathematically equivalent to  $\sqrt{x}$ .
- (B)  $x$  was imprecisely calculated in a previous program statement.
- (C) The computer stores floating-point numbers with 32-bit words.
- (D) There is round-off error in calculating the `pow` and `sqrt` functions.
- (E) There is overflow error in calculating the `pow` function.

7. Consider the following code segment

```
if (n != 0 && x/n > 100)  
    statement1;  
else  
    statement2;
```

If  $n$  is of type `int` and has a value of 0 when the segment is executed, what will happen?

- (A) An `ArithmeticException` will be thrown.
- (B) A syntax error will occur.
- (C) `statement1`, but not `statement2`, will be executed.
- (D) `statement2`, but not `statement1`, will be executed.
- (E) Neither `statement1` nor `statement2` will be executed; control will pass to the first statement following the `if` statement.

8. What will the output be for the following poorly formatted program segment, if the input value for `num` is 22?

```
int num = call to a method that reads an integer;  
if (num > 0)  
    if (num % 5 == 0)  
        System.out.println(num);  
    else System.out.println(num + " is negative");
```

- (A) 22
- (B) 4
- (C) 2 is negative
- (D) 22 is negative
- (E) Nothing will be output.

10. What values are stored in `x` and `y` after execution of the following program segment?

```
int x=30, y=40;  
if (x >= 0)  
{  
    if (x <= 100)  
    {  
        y = x*3;  
        if (y < 50)  
            x /= 10;  
    }  
    else  
        y = x*2;  
}  
else  
    y = -x;
```

- (A) x=30 y=90
- (B) x=30 y=-30
- (C) x=30 y=60
- (D) x=3 y=-3
- (E) x=30 y=40

16. Which *best* describes method `Mystery`?

```
int Mystery(int x, int y)  
//Precondition: x > y  
{  
    int i = 1, m = x;  
    while (m % y != 0)  
    {  
        i++;  
        m = i*x;  
    }  
    return m;  
}
```

- (A) It returns the smallest common factor of  $x$  and  $y$ , that is, the smallest positive integer divisor of both  $x$  and  $y$ .
- (B) It returns the greatest common factor of  $x$  and  $y$ , that is, the largest integer divisor of both  $x$  and  $y$ .
- (C) It returns the least common multiple of  $x$  and  $y$ , that is, the smallest integer that has both  $x$  and  $y$  as a factor.
- (D) It returns  $y$  raised to the  $x$ th power, that is,  $y^x$ .
- (E) It returns  $x$  raised to the  $y$ th power, that is,  $x^y$ .

```

//Precondition: n > 0
//Postcondition: returns n with its digits reversed
//Example: If n = 234, method reverse returns 432
int reverse(int n)
{
    int rem, revNum=0;

    <code segment>

    return revNum;
}

```

Which of the following replacements for <code segment> would cause the method to work as intended?

```

I for (int i=0; i<=n; i++)
{
    rem = n % 10;
    revNum = revNum*10 + rem;
    n /= 10;
}

II while (n != 0)
{
    rem = n % 10;
    revNum = revNum*10 + rem;
    n /= 10;
}

III for (int i = n; i != 0; i /= 10)
{
    rem = i % 10;
    revNum = revNum*10 + rem;
}

```

- (A) I only
- (B) II only
- (C) I and II only
- (D) II and III only
- (E) I and III only