

310-019: Sun Certified Java Associate for Java Edition 6

1

Click the Exhibit button.

What is the result?

- A. 0
- B. 2
- C. 4
- D. 6
- E. 9
- F. 13

```
1. public class Test {
2.     public static void main(String args[])
3.     {
4.         int i = 1, j = 0;
5.         switch(i) {
6.             case 2: j += 6;
7.             case 4: j += 1;
8.             default: j += 2;
9.             case 0: j += 4;
10.        }
11.        System.out.println("j = " + j);
12.    }
```

2

Which fragment is an example of inappropriate use of assertions?

- A.

```
assert (!(map.contains(x)));
map.add(x);
```
- B.

```
if (x > 0) {
} else {
    assert (x == 0);
}
```
- C.

```
public void aMethod(int x) {
    assert (x > 0);
}
```
- D.

```
assert (invariantCondition());
return retval;
```
- E.

```
switch (x) {
    case 1: break;
    case 2: break;
    default: assert (x == 0);
}
```

3

Click the Exhibit button.

What is the result?

- A. ABCD
- B. Compilation fails.
- C. c is printed before exiting with an error message.
- D. bc is printed before exiting with an error message.
- E. bcd is printed before exiting with an error message.

```
1. public class X {
2.     public static void main(String [] args)
3.     {
4.         try {
5.             badMethod();
6.             System.out.print("A");
7.         }
8.         catch (Exception ex) {
9.             System.out.print("B");
10.        }
11.        finally {
12.            System.out.print("C");
13.        }
14.        System.out.print("D");
15.    }
16.    public static void badMethod() {
17.        throw new Error();
18.    }
19. }
```

4

Given:

```
11. try {
12.     int x = 0;
13.     int y = 5 / x;
14. } catch (Exception e) {
15.     System.out.println("Exception");
16. } catch (ArithmeticException ae) {
17.     System.out.println("Arithmetic Exception");
18. }
19. System.out.println("finished");
```

What is the result?

- A. finished
- B. Exception
- C. Compilation fails.
- D. Arithmetic Exception

5

Click the Exhibit button.

What is the result?

- A. 0000000000
- B. 0123456789
- C. Compilation fails.
- D. The code runs with no output.
- E. The code enters an infinite loop.
- F. An exception is thrown at runtime.

```
1. public class Alpha1 {
2.     public static void main( String[] args
3. ) {
4.     boolean flag; int i=0;
5.     do {
6.         flag = false;
7.         System.out.println( i++ );
8.         flag = i < 10;
9.         continue;
10.    } while ( (flag)? true:false );
11. }
12. }
```

6

Given:

```
1. public class Foo {
2.     public static void main(String[] args) {
3.         try {
4.             return;
5.         } finally {
6.             System.out.println( "Finally" );
7.         }
8.     }
9. }
```

What is the result?

- A. Finally
- B. Compilation fails.
- C. The code runs with no output.
- D. An exception is thrown at runtime.

7

Given:

```
11. public class Test {
12.     public void foo() {
13.         assert false ;
14.         assert false ;
15.     }
16.     public void bar() {
17.         while(true) {
18.             assert false ;
19.         }
20.         assert false ;
21.     }
22. }
```

What causes compilation to fail?

- A. Line 13
- B. Line 14
- C. Line 18
- D. Line 20

8

Given:

```
11.     int i = 0, j = 5;
12. tp: for (;;) {
13.     i++;
14.     for (;;) {
15.         if (i > --j) {
16.             break tp;
17.         }
18.     }
19. }
20.     System.out.println("i = " + i + ", j = " + j);
```

What is the result?

- A. i = 1, j = 0
- B. i = 1, j = 4
- C. i = 3, j = 4
- D. i = 3, j = 0
- E. Compilation fails.

9

Given:

```
11. int x = 3;
12. int y = 1;
13. if (x = y) {
14.     System.out.println("x = " + x);
15. }
```

What is the result?

- A. x = 1
- B. x = 3
- C. Compilation fails.
- D. The code runs with no output.
- E. An exception is thrown at runtime.

10

Click the Exhibit button.

What is the result?

- A. T1 T2 T1 T2 T1 T2 T1 T2
- B. T1 T1 T1 T1 T2 T2 T2 T2
- C. T2 T2 T2 T2 T1 T1 T1 T1
- D. Deadlock; there is no output.
- E. The behavior is not specified by the Java Language Specification.

```
1. class ThreadTest extends Thread(
2.     int n ;
3.     public ThreadTest( int n ) { this.n = n
; }
4.     public void run(){
5.         for(int i=0;i<3;i++)
6.             System.out.print("T"+n+" ") ;
7.     }
8.     public static void main( String[] args
) {
9.         Thread t1 = new ThreadTest(1) ;
10.        Thread t2 = new ThreadTest(2) ;
12.        t1.setPriority( Thread.MAX_PRIORITY )
;
13.        t2.setPriority( Thread.MIN_PRIORITY )
;
14.        t1.start();
15.        t2.start() ;
16.    }
17. }
```