Oracle_1Z0-851_v2012-08-25_by_JK

Number: 1Z0-851
Passing Score: 610
Time Limit: 150 min
File Version: 2012-08-26

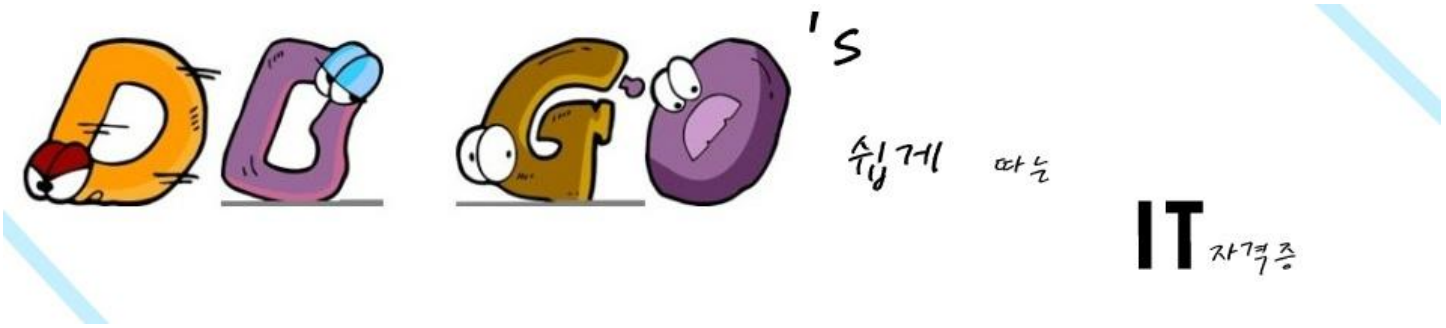
Exam : Oracle 1Z0-851

Version : 2012-08-26

Questions : 238

by : JK

Sections
1. All



Exam A

QUESTION 1
Which two statements are true? (Choose two.)

A. It is possible for more than two threads to deadlock at once.
B. The JVM implementation guarantees that multiple threads cannot enter into a deadlocked state.
C. Deadlocked threads release once their `sleep()` method's sleep duration has expired.
D. Deadlocking can occur only when the `wait()`, `notify()`, and `notifyAll()` methods are used incorrectly.
E. It is possible for a single-threaded application to deadlock if synchronized blocks are used incorrectly.
F. If a piece of code is capable of deadlocking, you cannot eliminate the possibility of deadlocking by inserting invocations of `Thread.yield()`.

Answer: AF
Section: All

Explanation/Reference:

QUESTION 2
Given:

```
void waitForSignal() {
    Object obj = new Object();
    synchronized (Thread.currentThread()) {
        obj.wait();
        obj.notify();
    }
}
```

Which statement is true?

A. This code can throw an `InterruptedException`.
B. This code can throw an `IllegalMonitorStateException`.
C. This code can throw a `TimeoutException` after ten minutes.
D. Reversing the order of `obj.wait()` and `obj.notify()` might cause this method to complete normally.
E. A call to `notify()` or `notifyAll()` from another thread might cause this method to complete normally.
F. This code does NOT compile unless `obj.wait()` is replaced with `((Thread) obj).wait()`.

Answer: A
Section: All

Explanation/Reference:

```
unreported exception java.lang.InterruptedException; must be caught or declared to be thrown
            obj.wait();
                   ^
1 error
```

QUESTION 3
What is the output if the main() method is run?

```
1. public class Starter extends Thread {
2.     private int x = 2;
3.     public static void main(String[] args) throws Exception {
4.         new Starter().makeItSo();
5.     }
6.     public Starter(){
7.         x = 5;
8.         start();
9.     }
10.    public void makeItSo() throws Exception {
11.        join();
12.        x = x - 1;
13.        System.out.println(x);
14.    }
15.    public void run() { x *= 2; }
16.}
```

A. 4
B. 5
C. 8
D. 9
E. Compilation fails.
F. An exception is thrown at runtime.
G. It is impossible to determine for certain.

Answer: D
Section: All

Explanation/Reference:
9

QUESTION 4
Given:

```
1. class PingPong2 {
2.     synchronized void hit(long n) {
3.         for(int i = 1; i < 3; i++)
4.             System.out.print(n + "-" + i + " ");
5.     }
6. }
```

```
1. public class Tester implements Runnable {
2.     static PingPong2 pp2 = new PingPong2();
3.     public static void main(String[] args) {
4.         new Thread(new Tester()).start();
5.         new Thread(new Tester()).start();
6.     }
7.     public void run() { pp2.hit(Thread.currentThread().getId()); }
8. }
```

Which statement is true?

A. The output could be `5-1 6-1 6-2 5-2`
B. The output could be `6-1 6-2 5-1 5-2`
C. The output could be `6-1 5-2 6-2 5-1`
D. The output could be `6-1 6-2 5-1 7-1`

Answer: B
Section: All

Explanation/Reference:
Since hit method is synchronized the answer for this question will be
`n-1 n-2 m-1 m-2`

QUESTION 5
Given:

```
1. public abstract class Shape {
2.     private int x;
3.     private int y;
4.     public abstract void draw();
5.     public void setAnchor(int x, int y) {
6.         this.x = x;
7.         this.y = y;
8.     }
9. }
```

Which two classes use the Shape class correctly? (Choose two.)

A.
```
public class Circle implements Shape {
    private int radius;
}
```
B.
```
public abstract class Circle extends Shape {
    private int radius;
}
```
C.
```
public class Circle extends Shape {
    private int radius;
    public void draw();
}
```
D.
```
public abstract class Circle implements Shape {
    private int radius;
    public void draw();
}
```
E.
```
public class Circle extends Shape {
    private int radius;
    public void draw() {/* code here */}
}
```
F.
```
public abstract class Circle implements Shape {
    private int radius;
    public void draw() {/* code here */}
}
```

Answer: BE
Section: All

Explanation/Reference:

QUESTION 6
Given:

```
1. public class Barn {
2.     public static void main(String[] args) {
3.         new Barn().go("hi", 1);
4.         new Barn().go("hi", "world", 2);
5.     }
6.     public void go(String... y, int x) {
7.         System.out.print(y[y.length - 1] + " ");
8.     }
9. }
```

What is the result?

A. `hi hi`
B. `hi world`
C. `world world`
D. Compilation fails.
E. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:

```
Main.java:6: ')' expected
    public void go(String... y, int x) {
                           ^
Main.java:6: ';' expected
    public void go(String... y, int x) {
                                ^
2 errors
```

The variable argument type String of the method go must be the last parameter and it isn't.

QUESTION 7
Given:

```
09. class Nav{
10.    public enum Direction { NORTH, SOUTH, EAST, WEST }
11. }
12.
13. public class Sprite{
14.     //    insert code here
15. }
```

Which code, inserted at line 14, allows the `Sprite` class to compile?

A. `Direction d = NORTH;`
B. `Nav.Direction d = NORTH;`
C. `Direction d = Direction.NORTH;`
D. `Nav.Direction d = Nav.Direction.NORTH;`

Answer: D
Section: All

Explanation/Reference:
Enum is accessible via `<class name>.<enum name>`.

Its values are accessible via `<class name>.<enum name>.<enum value>`.

QUESTION 8
Which statement is true about the classes and interfaces?

```
1. public interface A {
2.     public void doSomething(String thing);
3. }
```

```
1. public class AImpl implements A {
2.     public void doSomething(String msg) {}
3. }
```

```
1. public class B {
2.     public A doit(){
3.         //more code here
4.     }
5.     public String execute(){
6.         //more code here
7.     }
8. }
```

```
1. public class C extends B {
2.     public AImpl doit(){
3.         //more code here
4.     }
5.
6.     public Object execute() {
7.         //more code here
8.     }
9. }
```

A. Compilation will succeed for all classes and interfaces.
B. Compilation of class `C` will fail because of an error in line 2.
C. Compilation of class `C` will fail because of an error in line 6.
D. Compilation of class `AImpl` will fail because of an error in line 2.

Answer: C
Section: All

Explanation/Reference:

```
C.java:6: execute() in C cannot override execute() in B; attempting to use incompatible return
type
found   : java.lang.Object
required: java.lang.String
    public Object execute() {
```

There is a compilation error in class `C` (Line 6) since `execute()` method is not correctly overriding Class `B execute()` method.

QUESTION 9
What is the result?

```
11. public class Person {
12.    String name = "No name";
13.    public Person(String nm) { name = nm; }
```

```
14. }
15.
16. public class Employee extends Person {
17.    String empID = "0000";
18.    public Employee(String id) { empID = id; }
19. }
20.
21. public class EmployeeTest {
22.    public static void main(String[] args){
23.       Employee e = new Employee("4321");
24.       System.out.println(e.empID);
25.    }
26. }
```

A. `4321`
B. `0000`
C. An exception is thrown at runtime.
D. Compilation fails because of an error in line 18.

Answer: D
Section: All

Explanation/Reference:

```
Main.java:18: cannot find symbol
symbol  : constructor Person()
location: class Person
   public Employee(String id) { empID = id; }
                           ^
1 error
```

Implicit `super` constructor `Person()` is undefined. Must explicitly invoke another constructor

QUESTION 10
Given:

```
1. public class Rainbow {
2.    public enum MyColor {
3.       RED(0xff0000), GREEN(0x00ff00), BLUE(0x0000ff);
4.       private final int rgb;
5.       MyColor(int rgb) { this.rgb = rgb; }
6.       public int getRGB() { return rgb; }
7.    };
8.    public static void main(String[] args) {
9.       //insert code here
10.   }
11.}
```

Which code fragment, inserted at line 19, allows the `Rainbow` class to compile?

A. `MyColor skyColor = BLUE;`
B. `MyColor treeColor = MyColor.GREEN;`
C. `if(RED.getRGB() < BLUE.getRGB()) { }`
D. Compilation fails due to other error(s) in the code.
E. `MyColor purple = new MyColor(0xff00ff);`
F. `MyColor purple = MyColor.BLUE + MyColor.RED;`

Answer: B
Section: All

Explanation/Reference:
A.
```
Main.java:9: cannot find symbol
symbol  : variable BLUE
location: class Rainbow
       MyColor skyColor = BLUE;
                          ^
1 error
```

B.
```
Compiled successfully
```

C.
```
Main.java:9: cannot find symbol
symbol  : variable RED
location: class Rainbow
       if(RED.getRGB() < BLUE.getRGB()) { }
          ^
Main.java:9: cannot find symbol
symbol  : variable BLUE
location: class Rainbow
       if(RED.getRGB() < BLUE.getRGB()) { }
                         ^
2 errors
```

D.
```
Other code compiled successfully. This option is not correct.
```

E.
```
Main.java:9: enum types may not be instantiated
       MyColor purple = new MyColor(0xff00ff);
                        ^
1 error
```

F.
```
Main.java:9: operator + cannot be applied to Rainbow.MyColor,Rainbow.MyColor
       MyColor purple = MyColor.BLUE + MyColor.RED;
                                     ^
1 error
```

QUESTION 11
Given:

```
1. class Atom {
2.     Atom() { System.out.print("atom "); }
3. }
4. class Rock extends Atom {
5.     Rock(String type) { System.out.print(type); }
6. }
7. public class Mountain extends Rock {
8.     Mountain() {
9.         super("granite ");
10.        new Rock("granite ");
11.    }
12.    public static void main(String[] a) { new Mountain(); }
13.}
```

What is the result?

A. Compilation fails.
B. `atom granite`
C. `granite granite`
D. `atom granite granite`
E. An exception is thrown at runtime.
F. `atom granite atom granite`

Answer: F
Section: All

Explanation/Reference:
`atom granite atom granite`

QUESTION 12
Given:

```
01. interface TestA { String toString(); }
02.
03. public class Test {
04.     public static void main(String[] args) {
05.         System.out.println(new TestA() {
06.             public String toString() { return "test"; }
07.         });
08.     }
09. }
```

What is the result?

A. `test`
B. `null`
C. An exception is thrown at runtime.
D. Compilation fails because of an error in line 1.
E. Compilation fails because of an error in line 4.
F. Compilation fails because of an error in line 5.

Answer: A
Section: All

Explanation/Reference:
`test`

QUESTION 13
Given:

```
1.     public static void parse(String str) {
2.         try {
3.             float f = Float.parseFloat(str);
4.         } catch (NumberFormatException nfe) {
5.             f = 0;
6.         } finally {
7.             System.out.println(f);
8.         }
9.     }
10.    public static void main(String[] args) {
```

```
11.        parse("invalid");
12.    }
```
What is the result?

A. `0.0`
B. Compilation fails.
C. A `ParseException` is thrown by the `parse` method at runtime.
D. A `NumberFormatException` is thrown by the `parse` method at runtime.

Answer: B
Section: All

Explanation/Reference:
```
Main.java:5: cannot find symbol
symbol  : variable f
location: class Test
          f = 0;
          ^
Main.java:7: cannot find symbol
symbol  : variable f
location: class Test
          System.out.println(f);
                             ^
2 errors
```

`f` variable is being declared inside `try` block. Because of it, cannot be used neither inside `catch` nor `finally` blocks.

QUESTION 14
Given:

```
1. public class Blip {
2.     protected int blipvert(int x) { return 0; }
3. }
4. class Vert extends Blip {
5.     //    insert code here
6. }
```

Which five methods, inserted independently at line 5, will compile? (Choose five.)

A. `public int blipvert(int x) { return 0; }`
B. `private int blipvert(int x) { return 0; }`
C. `private int blipvert(long x) { return 0; }`
D. `protected long blipvert(int x) { return 0; }`
E. `protected int blipvert(long x) { return 0; }`
F. `protected long blipvert(long x) { return 0; }`
G. `protected long blipvert(int x, int y) { return 0; }`

Answer: ACEFG
Section: All

Explanation/Reference:
A. `public int blipvert(int x) { return 0; }`
```
compiled successfully
```

B. `private int blipvert(int x) { return 0; }`
```
Main.java:5: blipvert(int) in Vert cannot override blipvert(int) in Blip; attempting to assign
weaker access privileges; was protected
```

```
   private int blipvert(int x) { return 0; }
           ^
1 error
```

C.
```
private int blipvert(long x) { return 0; }
compiled successfully
```

D.
```
protected long blipvert(int x) { return 0; }
Main.java:5: blipvert(int) in Vert cannot override blipvert(int) in Blip; attempting to use
incompatible return type
found   : long
required: int
   protected long blipvert(int x) { return 0; }
                ^
1 error
```

E.
```
protected int blipvert(long x) { return 0; }
compiled successfully
```

F.
```
protected long blipvert(long x) { return 0; }
compiled successfully
```

G.
```
protected long blipvert(int x, int y) { return 0; }
compiled successfully
```

QUESTION 15
Given:

```
1. class Super {
2.    private int a;
3.    protected Super(int a) { this.a = a; }
4. }

11. class Sub extends Super {
12.    public Sub(int a) { super(a); }
13.    public Sub() { this.a = 5; }
14. }
```

Which two, independently, will allow Sub to compile? (Choose two.)

A. Change line 2 to:
   public int a;
B. Change line 2 to:
   protected int a;
C. Change line 13 to:
   public Sub() { this(5); }
D. Change line 13 to:
   public Sub() { super(5); }
E. Change line 13 to:
   public Sub() { super(a); }

Answer: CD
Section: All

Explanation/Reference:
A.
```
Main.java:13: cannot find symbol
symbol  : constructor Super()
location: class Super
```

```
    public Sub() { this.a = 5; }
                ^
1 error
```

B.

```
Main.java:13: cannot find symbol
symbol  : constructor Super()
location: class Super
    public Sub() { this.a = 5; }
                ^
1 error
```

C.

```
compiled successfully.
```

D.

```
compiled successfully.
```

E.

```
Main.java:13: a has private access in Super
    public Sub() { super(a); }
                       ^
1 error
```

QUESTION 16
Which `Man` class properly represents the relationship "Man has a best friend who is a Dog"?

A. `class Man extends Dog { }`
B. `class Man implements Dog { }`
C. `class Man { private BestFriend dog; }`
D. `class Man { private Dog bestFriend; }`
E. `class Man { private Dog<bestFriend>; }`
F. `class Man { private BestFriend<dog>; }`

Answer: D
Section: All

Explanation/Reference:
1) Has-a is implemented using instance variables. (`A` and `B` are excluded answers)
2) best friend who is-a Dog -> the variable `bestFriend` has to be from `Dog` class. (`C`, `E` and `F` are excluded answers)

QUESTION 17
Given:

```
1. package test;
2.
3. class Target {
4.     public String name = "hello";
5. }
```

What can directly access and change the value of the variable name?

A. any class
B. only the `Target` class
C. any class in the `test` package
D. any class that extends `Target`

Answer: C
Section: All

Explanation/Reference:

QUESTION 18
Given:

```
11. abstract class Vehicle { public int speed() { return 0; }
12. class Car extends Vehicle { public int speed() { return 60; }
13. class RaceCar extends Car { public int speed() { return 150; } ...

21.    RaceCar racer = new RaceCar();
22.    Car car = new RaceCar();
23     Vehicle vehicle = new RaceCar();
24     System.out.println(racer.speed() + ", " + car.speed() + ", " + vehicle.speed());
```

What is the result?

A. `0, 0, 0`
B. `150, 60, 0`
C. Compilation fails.
D. `150, 150, 150`
E. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:
`150, 150, 150`

QUESTION 19
Given:

```
5. class Building { }
6. public class Barn extends Building {
7.    public static void main(String[] args) {
8.        Building build1 = new Building();
9.        Barn barn1 = new Barn();
10.       Barn barn2 = (Barn) build1;
11.       Object obj1 = (Object) build1;
12.       String str1 = (String) build1;
13.       Building build2 = (Building) barn1;
14.   }
15. }
```

Which is true?

A. If line 10 is removed, the compilation succeeds.
B. If line 11 is removed, the compilation succeeds.
C. If line 12 is removed, the compilation succeeds.
D. If line 13 is removed, the compilation succeeds.
E. More than one line must be removed for compilation to succeed.

Answer: C

Section: All

Explanation/Reference:

```
Main.java:12: inconvertible types
found   : Building
required: java.lang.String
      String str1 = (String) build1;
                             ^
1 error
```

Cannot cast from `Building` to `String`

QUESTION 20
A team of programmers is reviewing a proposed API for a new utility class. After some discussion, they realize that they can reduce the number of methods in the API without losing any functionality. If they implement the new design, which two OO principles will they be promoting?

A. Looser coupling
B. Tighter coupling
C. Lower cohesion
D. Higher cohesion
E. Weaker encapsulation
F. Stronger encapsulation

Answer: AD
Section: All

Explanation/Reference:

QUESTION 21
Given:

```
21. class Money {
22.    private String country = "Canada";
23.    public String getC() { return country; }
24. }
25. class Yen extends Money {
26.    public String getC() { return super.country; }
27. }
28. public class Euro extends Money {
29.    public String getC(int x) { return super.getC(); }
30.    public static void main(String[] args) {
31.       System.out.print(new Yen().getC() + " " + new Euro().getC());
32.    }
33. }
```

What is the result?

A. `Canada`
B. `null Canada`
C. `Canada null`
D. `Canada Canada`
E. Compilation fails due to an error on line 26.
F. Compilation fails due to an error on line 29.

Answer: E

Section: All

Explanation/Reference:
```
Yen.java:26: country has private access in Money
   public String getC() { return super.country; }
                                      ^
1 error
```

The field `Money.country` is not visible (`Money`'s private var)

QUESTION 22
Assuming that the serializeBanana() and the deserializeBanana() methods will correctly use Java serialization and given:

```
13. import java.io.*;
14. class Food implements Serializable {int good = 3;}
15. class Fruit extends Food {int juice = 5;}
16. public class Banana extends Fruit {
17.    int yellow = 4;
18.    public static void main(String [] args) {
19.       Banana b = new Banana(); Banana b2 = new Banana();
20.       b.serializeBanana(b); // assume correct serialization
21.       b2 = b.deserializeBanana(); // assume correct
22.       System.out.println("restore "+b2.yellow+ b2.juice+b2.good);
24.    }
25. //    more Banana methods go here
50. }
```

What is the result?

A. `restore 400`
B. `restore 403`
C. `restore 453`
D. Compilation fails.
E. An exception is thrown at runtime.

Answer: C
Section: All

Explanation/Reference:
```
restore 453
```

QUESTION 23
Given a valid `DateFormat` object named `df`, and

```
16. Date d = new Date(0L);
17. String ds = "December 15, 2004";
18. //insert code here
```

What updates `d`'s value with the date represented by `ds`?

A.
```
18. d = df.parse(ds);
```
B.
```
18. d = df.getDate(ds);
```
C.
```
18. try {
19. d = df.parse(ds);
20. } catch(ParseException e) { };
```
D.
```
18. try {
19. d = df.getDate(ds);
```

```
20. } catch(ParseException e) { };
```

Answer: C
Section: All

Explanation/Reference:

QUESTION 24
Given:

```
11. double input = 314159.26;
12. NumberFormat nf = NumberFormat.getInstance(Locale.ITALIAN);
13. String b;
14. //insert code here
```

Which code, inserted at line 14, sets the value of `b` to `314.159,26`?

A. `b = nf.parse( input );`
B. `b = nf.format( input );`
C. `b = nf.equals( input );`
D. `b = nf.parseObject( input );`

Answer: B
Section: All

Explanation/Reference:
A.

```
Main.java:14: cannot find symbol
symbol  : method parse(double)
location: class java.text.NumberFormat
   b = nf.parse( input );
         ^
1 error
```

B. `314.159,26`

C.

```
Main.java:14: incompatible types
found   : boolean
required: java.lang.String
   b = nf.equals( input );
              ^
1 error
```

D.

```
Main.java:14: cannot find symbol
symbol  : method parseObject(double)
location: class java.text.NumberFormat
   b = nf.parseObject( input );
         ^
1 error
```

QUESTION 25
Given:

```
1. public class TestString1 {
2.     public static void main(String[] args) {
```

```
3.         String str = "420";
4.         str += 42;
5.         System.out.print(str);
6.     }
7. }
```

What is the output?

A. 42
B. 420
C. 462
D. 42042
E. Compilation fails.
F. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:
42042

QUESTION 26
Which capability exists only in `java.io.FileWriter`?

A. Closing an open stream.
B. Flushing an open stream.
C. Writing to an open stream.
D. Writing a line separator to an open stream.

Answer: D
Section: All

Explanation/Reference:

QUESTION 27
Given that the current directory is empty, and that the user has read and write permissions, and the following:

```
1. import java.io.*;
2. public class DOS {
3.     public static void main(String[] args) {
4.         File dir = new File("dir");
5.         dir.mkdir();
6.         File f1 = new File(dir, "f1.txt");
7.         try {
8.             f1.createNewFile();
9.         } catch (IOException e) { ; }
10.        File newDir = new File("newDir");
11.        dir.renameTo(newDir);
12.    }
13.}
```

Which statement is true?

A. Compilation fails.

B. The file system has a new empty directory named `dir`.
C. The file system has a new empty directory named `newDir`.
D. The file system has a directory named `dir`, containing a file f1.txt.
E. The file system has a directory named `newDir`, containing a file `f1.txt`.

Answer: E
Section: All

Explanation/Reference:

QUESTION 28
Given:

```
1. public class Score implements Comparable<Score> {
2.     private int wins, losses;
3.     public Score(int w, int l) { wins = w; losses = l; }
4.     public int getWins() { return wins; }
5.     public int getLosses() { return losses; }
6.     public String toString() {
7.         return "<" + wins + "," + losses + ">";
8.     }
9. //    insert code here
10.}
```

Which method will complete this class?

A. `public int compareTo(Object o){/*more code here*/}`
B. `public int compareTo(Score other){/*more code here*/}`
C. `public int compare(Score s1,Score s2){/*more code here*/}`
D. `public int compare(Object o1,Object o2){/*more code here*/}`

Answer: B
Section: All

Explanation/Reference:
A.

```
Main.java:1: Score is not abstract and does not override abstract method compareTo(Score) in
java.lang.Comparable
class Score implements Comparable<Score> {
^
1 error
```

B.

```
compiled successfully
```

C.

```
Main.java:1: Score is not abstract and does not override abstract method compareTo(Score) in
java.lang.Comparable
class Score implements Comparable<Score> {
^
1 error
```

D.

```
Main.java:1: Score is not abstract and does not override abstract method compareTo(Score) in
java.lang.Comparable
class Score implements Comparable<Score> {
^
```

```
1 error
```

QUESTION 29
Given:

```
22. StringBuilder sb1 = new StringBuilder("123");
23. String s1 = "123";
24. // insert code here
25. System.out.println(sb1 + " " + s1);
```

Which code fragment, inserted at line 24, outputs `123abc 123abc`?

A. `sb1.append("abc"); s1.append("abc");`
B. `sb1.append("abc"); s1.concat("abc");`
C. `sb1.concat("abc"); s1.append("abc");`
D. `sb1.concat("abc"); s1.concat("abc");`
E. `sb1.append("abc"); s1 = s1.concat("abc");`
F. `sb1.concat("abc"); s1 = s1.concat("abc");`
G. `sb1.append("abc"); s1 = s1 + s1.concat("abc");`
H. `sb1.concat("abc"); s1 = s1 + s1.concat("abc");`

Answer: E
Section: All

Explanation/Reference:
A.

```
Main.java:24: cannot find symbol
symbol  : method append(java.lang.String)
location: class java.lang.String
sb1.append("abc"); s1.append("abc");
                   ^
1 error
```

B. 123abc 123

C.

```
Main.java:24: cannot find symbol
symbol  : method concat(java.lang.String)
location: class java.lang.StringBuilder
sb1.concat("abc"); s1.append("abc");
   ^
Main.java:24: cannot find symbol
symbol  : method append(java.lang.String)
location: class java.lang.String
sb1.concat("abc"); s1.append("abc");
                   ^
2 errors
```

D.

```
Main.java:5: cannot find symbol
symbol  : method concat(java.lang.String)
location: class java.lang.StringBuilder
sb1.concat("abc"); s1.concat("abc");
   ^
1 error
```

E. 123abc 123abc

F.

```
Main.java:24: cannot find symbol
symbol  : method concat(java.lang.String)
location: class java.lang.StringBuilder
sb1.concat("abc"); s1 = s1.concat("abc");
   ^
1 error
```

G. 123abc 123123abc

H.
```
Main.java:24: cannot find symbol
symbol  : method concat(java.lang.String)
location: class java.lang.StringBuilder
sb1.concat("abc"); s1 = s1 + s1.concat("abc");
   ^
1 error
```

QUESTION 30
Click the Exhibit button.
Which code, inserted at line 14, will allow this class to correctly serialize and deserialize?

```
1. import java.io.*;
2. public class Foo implements Serializable {
3.     public int x, y;
4.     public Foo(int x, int y){
5.         this.x = x; this.y = y;
6.     }
7.
8.     private void writeObject(ObjectOutputStream s)
9.         throws IOException{
10.        s.writeInt(x); s.writeInt(y);
11.    }
12.
13.    private void readObject(ObjectInputStream s)
14.        throws IOException, ClassNotFoundException {
15.        //insert code here
16.    }
17. }
```

A. `s.defaultReadObject();`
B. `this = s.defaultReadObject();`
C. `y = s.readInt(); x = s.readInt();`
D. `x = s.readInt(); y = s.readInt();`

Answer: D
Section: All

Explanation/Reference:

QUESTION 31
Given:
```
interface Foo {}
class Alpha implements Foo {}
class Beta extends Alpha {}
```

```
class Delta extends Beta {
   public static void main( String[] args ) {
      Beta x = new Beta();
16.  //insert code here
   }
}
```

Which code, inserted at line 16, will cause a java.lang.ClassCastException?

A. `Alpha a = x;`
B. `Foo f = (Delta)x;`
C. `Foo f = (Alpha)x;`
D. `Beta b = (Beta)(Alpha)x;`

Answer: B
Section: All

Explanation/Reference:
A.
```
compiled successfully
```

B.
```
Exception in thread "main" java.lang.ClassCastException: Beta cannot be cast to Delta
   at Delta.main(Main.java:16)
```

C.
```
compiled successfully
```

D.
```
compiled successfully
```

QUESTION 32
Given:

```
public void go() {
   String o = "";
   z:
      for(int x = 0; x < 3; x++) {
         for(int y = 0; y < 2; y++) {
            if(x==1) break;
            if(x==2 && y==1) break z;
            o = o + x + y;
         }
      }
      System.out.println(o);
}
```

What is the result when the `go()` method is invoked?

A. `00`
B. `0001`
C. `000120`
D. `00012021`
E. Compilation fails.
F. An exception is thrown at runtime.

Answer: C
Section: All

Explanation/Reference:
000120

QUESTION 33
Given:

```
33. try {
34.  //some code here
35. } catch (NullPointerException e1) {
36.    System.out.print("a");
37. } catch (Exception e2) {
38.    System.out.print("b");
39. } finally {
40.    System.out.print("c");
41. }
```

If some sort of exception is thrown at line 34, which output is possible?

A. a
B. b
C. c
D. ac
E. abc

Answer: D
Section: All

Explanation/Reference:
ac
bc

QUESTION 34
Given:

```
31. //some code here
32. try {
33.  //some code here
34. } catch (NullPointerException e1) {
35.  //some code here
36. } finally {
37.  //some code here
38. }
```

Under which three circumstances will the code on line 37 be executed? (Choose three.)

A. The instance gets garbage collected.
B. The code on line 33 throws an exception.
C. The code on line 35 throws an exception.
D. The code on line 31 throws an exception.
E. The code on line 33 executes successfully.

Answer: BCE
Section: All

Explanation/Reference:

QUESTION 35
Given:

```
public class Donkey2 {
    public static void main(String[] args) {
        boolean assertsOn = true;
        assert (assertsOn) : assertsOn = true;
        if(assertsOn) {
            System.out.println("assert is on");
        }
    }
}
```

If class `Donkey` is invoked twice, the first time without assertions enabled, and the second time with assertions enabled, what are the results?

A. no output
B. no output
   `assert is on`
C. `assert is on`
D. no output
   An AssertionError is thrown.
E. assert is on
   An AssertionError is thrown.

Answer: C
Section: All

Explanation/Reference:

QUESTION 36
Given:

```
public void method() {
    A a = new A();
    a.method1();
}
```

Which statement is true if a TestException is thrown on line 3 of class B?

```
1. public class A{
2.     public void method1() {
3.         try {
4.             B b = new B();
5.             b.method2();
6.             //more code here
7.         } catch (TestException te){
8.             throw new RuntimeException(te);
9.         }
10.    }
11.}
```

```
1. public class B{
2.     public void method2() throws TestException {
3.         //more code here
4.     }
```

```
5. }
```

```
1. class TestException extends Exception {
2. }
```

A. Line 33 must be called within a `try` block.
B. The exception thrown by `method1` in class `A` is not required to be caught.
C. The method declared on line 31 must be declared to throw a `RuntimeException`.
D. On line 5 of class `A`, the call to `method2` of class `B` does not need to be placed in a `try`/`catch` block.

Answer: B
Section: All

Explanation/Reference:

QUESTION 37
Given:

```
01. Float pi = new Float(3.14f);
02. if (pi > 3) {
03.     System.out.print("pi is bigger than 3. ");
04. }
05. else {
06.     System.out.print("pi is not bigger than 3. ");
07. }
08. finally {
09.     System.out.println("Have a nice day.");
10. }
```

What is the result?

A. Compilation fails.
B. `pi is bigger than 3.`
C. An exception occurs at runtime.
D. `pi is bigger than 3. Have a nice day.`
E. `pi is not bigger than 3. Have a nice day.`

Answer: A
Section: All

Explanation/Reference:

```
Main.java:08: 'finally' without 'try'
finally {
^
1 error
```

QUESTION 38
Given:

```
1. public class Boxer1{
2.     Integer i;
3.     int x;
4.     public Boxer1(int y) {
```

```
5.         x = i+y;
6.         System.out.println(x);
7.     }
8.     public static void main(String[] args) {
9.         new Boxer1(new Integer(4));
10.    }
11. }
```

What is the result?

A. The value `4` is printed at the command line.
B. Compilation fails because of an error in line 5.
C. Compilation fails because of an error in line 9.
D. A `NullPointerException` occurs at runtime.
E. A `NumberFormatException` occurs at runtime.
F. An `IllegalStateException` occurs at runtime.

Answer: D
Section: All

Explanation/Reference:

```
Exception in thread "main" java.lang.NullPointerException
    at Boxer1.<init>(Main.java:5)
    at Boxer1.main(Main.java:9)
```

QUESTION 39
Given:

```
1. public class Person {
2.     private String name;
3.     public Person(String name) { this.name = name; }
4.     public boolean equals(Person p) {
5.         return p.name.equals(this.name);
6.     }
7. }
```

Which statement is true?

A. The `equals` method does NOT properly override the `Object.equals` method.
B. Compilation fails because the `private` attribute `p.name` cannot be accessed in line 5.
C. To work correctly with hash-based data structures, this class must also implement the `hashCode` method.
D. When adding `Person` objects to a `java.util.Set` collection, the `equals` method in line 4 will prevent duplicates.

Answer: A
Section: All

Explanation/Reference:

QUESTION 40
Which two statements are true about the `hashCode` method? (Choose two.)

A. The `hashCode` method for a given class can be used to test for object equality and object inequality for that class.
B. The `hashCode` method is used by the `java.util.SortedSet` collection class to order the elements within that set.
C. The `hashCode` method for a given class can be used to test for object inequality, but NOT object equality, for that class.
D. The only important characteristic of the values returned by a `hashCode` method is that the distribution of values must follow a Gaussian distribution.
E. The `hashCode` method is used by the `java.util.HashSet` collection class to group the elements within that set into hash buckets for swift retrieval.

Answer: CE
Section: All

Explanation/Reference:

QUESTION 41
Given a pre-generics implementation of a method:

```
11. public static int sum(List list) {
12.    int sum = 0;
13.    for ( Iterator iter = list.iterator(); iter.hasNext(); ) {
14.       int i = ((Integer)iter.next()).intValue();
15.       sum += i;
16.    }
17.    return sum;
18. }
```

What three changes allow the class to be used with generics and avoid an unchecked warning? (Choose three.)

A. Remove line 14.
B. Replace line 14 with `int i = iter.next();`
C. Replace line 13 with `for (int i : intList) {`
D. Replace line 13 with `for (Iterator iter : intList) {`
E. Replace the method declaration with `sum(List<int> intList)`
F. Replace the method declaration with `sum(List<Integer> intList)`

Answer: ACF
Section: All

Explanation/Reference:

```
public static int sum(List<Integer> intList) {
    int sum = 0;
    for (int i : intList) {
        sum += i;
    }
    return sum;
}
```

QUESTION 42
Given:

```
23. Object [] myObjects = {
24.       new Integer(12),
25.       new String("foo"),
```

```
26.       new Integer(5),
27.       new Boolean(true)
28. };
29. Arrays.sort(myObjects);
30. for(int i=0; i<myObjects.length; i++) {
31.    System.out.print(myObjects[i].toString());
32.    System.out.print(" ");
33. }
```

What is the result?

A. Compilation fails due to an error in line 23.
B. Compilation fails due to an error in line 29.
C. A `ClassCastException` occurs in line 29.
D. A `ClassCastException` occurs in line 31.
E. The value of all four objects prints in natural order.

Answer: C
Section: All

Explanation/Reference:

```
 Exception in thread "main" java.lang.ClassCastException: java.lang.String cannot be cast to
java.lang.Integer
    at java.lang.Integer.compareTo(Integer.java:37)
    at java.util.Arrays.mergeSort(Arrays.java:1144)
    at java.util.Arrays.sort(Arrays.java:1079)
    at main(Main.java:29)
```

QUESTION 43
Given a class Repetition:

```
1. package utils;
2.
3. public class Repetition {
4. public static String twice(String s) { return s + s; }
5. }
```

and given another class Demo:

```
1. public class Demo {
2.     public static void main(String[] args) {
3.         System.out.println(twice("pizza"));
4.     }
5. }
```

Which code should be inserted at line 1 of `Demo.java` to compile and run `Demo` to print `pizzapizza`?

A. `import utils.*;`
B. `static import utils.*;`
C. `import utils.Repetition.*;`
D. `static import utils.Repetition.*;`
E. `import utils.Repetition.twice();`
F. `import static utils.Repetition.twice;`
G. `static import utils.Repetition.twice;`

Answer: F
Section: All

Explanation/Reference:

QUESTION 44
A UNIX user named Bob wants to replace his chess program with a new one, but he is not sure where the old one is installed. Bob is currently able to run a Java chess program starting from his home directory /home/bob using the command:

```
java -classpath /test:/home/bob/downloads/*.jar games.Chess
```

Bob's CLASSPATH is set (at login time) to:

```
/usr/lib:/home/bob/classes:/opt/java/lib:/opt/java/lib/*.jar
```

What is a possible location for the Chess.class file?

A. `/test/Chess.class`
B. `/home/bob/Chess.class`
C. `/test/games/Chess.class`
D. `/usr/lib/games/Chess.class`
E. `/home/bob/games/Chess.class`
F. inside jarfile `/opt/java/lib/Games.jar` (with a correct manifest)
G. inside jarfile `/home/bob/downloads/Games.jar` (with a correct manifest)

Answer: C
Section: All

Explanation/Reference:

QUESTION 45
Given the following directory structure:

```
bigProject
|--source
| |--Utils.java
|
|--classes
|--
```

And the following command line invocation:

```
javac -d classes source/Utils.java
```

Assume the current directory is `bigProject`, what is the result?

A. If the compile is successful, `Utils.class` is added to the `source` directory.
B. The compiler returns an invalid flag error.
C. If the compile is successful, `Utils.class` is added to the `classes` directory.
D. If the compile is successful, `Utils.class` is added to the `bigProject` directory.

Answer: C
Section: All

Explanation/Reference:

QUESTION 46
Which statement is true?

A. A class's `finalize()` method CANNOT be invoked explicitly.
B. `super.finalize()` is called implicitly by any overriding `finalize()` method.
C. The `finalize()` method for a given object is called no more than once by the garbage collector.
D. The order in which `finalize()` is called on two objects is based on the order in which the two objects became finalizable.

Answer: C
Section: All

Explanation/Reference:

QUESTION 47
Given:

```
1. public class Batman {
2.     int squares = 81;
3.     public static void main(String[] args) {
4.         new Batman().go();
5.     }
6.     void go() {
7.         incr(++squares);
8.         System.out.println(squares);
9.     }
10.    void incr(int squares) { squares += 10; }
11. }
```

What is the result?

A. `81`
B. `82`
C. `91`
D. `92`
E. Compilation fails.
F. An exception is thrown at runtime.

Answer: B
Section: All

Explanation/Reference:
`squares` starts with `81`
`squares` value is incremented before entering in `incr` method (`82`).
inside `incr` method another `squares` is incremented by `10` (shadowing)
when `incr` method exits class variable `squares` continues being `82`.

`82`

QUESTION 48
Given:

```
public class Yippee {
    public static void main(String [] args) {
        for(int x = 1; x < args.length; x++) {
            System.out.print(args[x] + " ");
        }
    }
}
```

and two separate command line invocations:

```
java Yippee
java Yippee 1 2 3 4
```

What is the result?

A. No output is produced.
   `1 2 3`
B. No output is produced.
   `2 3 4`
C. No output is produced.
   `1 2 3 4`
D. An exception is thrown at runtime.
   `1 2 3`
E. An exception is thrown at runtime.
   `2 3 4`
F. An exception is thrown at runtime.
   `1 2 3 4`

Answer: B
Section: All

Explanation/Reference:

QUESTION 49
Given:

```
1. public class Pass {
2.     public static void main(String [] args) {
3.         int x = 5;
4.         Pass p = new Pass();
5.         p.doStuff(x);
6.         System.out.print(" main x = " + x);
7.     }
8.
9.     void doStuff(int x) {
10.        System.out.print(" doStuff x = " + x++);
11.    }
12. }
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. `doStuff x = 6 main x = 6`
D. `doStuff x = 5 main x = 5`
E. `doStuff x = 5 main x = 6`

F. `doStuff x = 6 main x = 5`

Answer: D
Section: All

Explanation/Reference:
`doStuff x = 5 main x = 5`

QUESTION 50
Given:

```
1. interface Animal { void makeNoise(); }
2. class Horse implements Animal {
3.     Long weight = 1200L;
4.     public void makeNoise() { System.out.println("whinny"); }
5. }
6.
7. public class Icelandic extends Horse {
8.     public void makeNoise() { System.out.println("vinny"); }
9.     public static void main(String[] args) {
10.        Icelandic i1 = new Icelandic();
11.        Icelandic i2 = new Icelandic();
12.        Icelandic i3 = new Icelandic();
13.        i3 = i1; i1 = i2; i2 = null; i3 = i1;
14.    }
15.}
```

When line 14 is reached, how many objects are eligible for the garbage collector?

A. 0
B. 1
C. 2
D. 3
E. 4
F. 6

Answer: E
Section: All

Explanation/Reference:

QUESTION 51
Given two files: `GrizzlyBear.java` and `Salmon.java`

```
1. package animals.mammals;
2.
3. public class GrizzlyBear extends Bear {
4. void hunt() {
5. Salmon s = findSalmon();
6. s.consume();
7. }
8. }
```

```
1. package animals.fish;
2.
3. public class Salmon extends Fish {
```

```
4. public void consume() { /* do stuff */ }
5. }
```

If both classes are in the correct directories for their packages, and the `Mammal` class correctly defines the `findSalmon()` method, which change allows this code to compile?

A. add `import animals.mammals.*;` at line 2 in `Salmon.java`
B. add `import animals.fish.*;` at line 2 in `GrizzlyBear.java`
C. add `import animals.fish.Salmon.*;` at line 2 in `GrizzlyBear.java`
D. add `import animals.mammals.GrizzlyBear.*;` at line 2 in `Salmon.java`

Answer: B
Section: All

Explanation/Reference:

QUESTION 52
Given:

```
String[] elements = { "for", "tea", "too" };
String first = (elements.length > 0) ? elements[0] : null;
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. The variable first is set to `null`.
D. The variable `first` is set to `elements[0]`.

Answer: D
Section: All

Explanation/Reference:
`"for" = elements[0]`

QUESTION 53
A company has a business application that provides its users with many different reports:
receivables reports, payables reports, revenue projects, and so on. The company has just purchased some new, state-of-the-art, wireless printers, and a programmer has been assigned the task of enhancing all of the reports to use not only the company's old printers, but the new wireless printers as well. When the programmer starts looking into the application, the programmer discovers that because of the design of the application, it is necessary to make changes to each report to support the new printers.Which two design concepts most likely explain this situation? (Choose two.)

A. Inheritance
B. Low cohesion
C. Tight coupling
D. High cohesion
E. Loose coupling
F. Object immutability

Answer: BC
Section: All

Explanation/Reference:

QUESTION 54
Given:

```
10. public class SuperCalc {
11.    protected static int multiply(int a, int b) { return a * b;}
12. }
and:
20.    public class SubCalc extends SuperCalc{
21.    public static int multiply(int a, int b) {
22.       int c = super.multiply(a, b);
23.       return c;
24.    }
25. }
and:
30. SubCalc sc = new SubCalc ();
31. System.out.println(sc.multiply(3,4));
32. System.out.println(SubCalc.multiply(2,2));
```

What is the result?

A. 12
B. The code runs with no output.
C. An exception is thrown at runtime.
D. Compilation fails because of an error in line 21.
E. Compilation fails because of an error in line 22.
F. Compilation fails because of an error in line 31.

Answer: E
Section: All

Explanation/Reference:

```
Main.java:22: non-static variable super cannot be referenced from a static context
       int c = super.multiply(a, b);
               ^
1 error
```

QUESTION 55
Given:

```
6. public class Threads2 implements Runnable {
7.
8.     public void run() {
9.        System.out.println("run.");
10.       throw new RuntimeException("Problem");
11.    }
12.    public static void main(String[] args) {
13.       Thread t = new Thread(new Threads2());
14.       t.start();
15.       System.out.println("End of method.");
16.    }
17.}
```

Which two can be results? (Choose two.)

A. java.lang.RuntimeException: Problem
B. run.
   java.lang.RuntimeException: Problem

C.
```
End of method.
java.lang.RuntimeException: Problem
```
D.
```
End of method.
run.
java.lang.RuntimeException: Problem
```
E.
```
run.
java.lang.RuntimeException: Problem
End of method.
```

Answer: DE
Section: All

Explanation/Reference:

D.
```
End of method.
run.
Exception in thread "Thread-0" java.lang.RuntimeException: Problem
       at Threads2.run(Threads2.java:5)
       at java.lang.Thread.run(Unknown Source)
```

E.
```
run.
Exception in thread "Thread-0" java.lang.RuntimeException: Problem
       at Threads2.run(Threads2.java:5)
       at java.lang.Thread.run(Unknown Source)
End of method.
```

QUESTION 56
Which two classes correctly implement both the `java.lang.Runnable` and the `java.lang.Cloneable` interfaces? (Choose two.)

A.
```
public class Session implements Runnable, Cloneable {
    public void run();

    public Object clone();
}
```
B.
```
public class Session extends Runnable, Cloneable {
    public void run() { /* do something */ }
    public Object clone() { /* make a copy */ }
}
```
C.
```
public class Session implements Runnable, Cloneable {
    public void run() { /* do something */ }
    public Object clone() { /* make a copy */ }
}
```
D.
```
public abstract class Session implements Runnable, Cloneable {
    public void run() { /* do something */ }
    public Object clone() { /*make a copy */ }
}
```
E.
```
public class Session implements Runnable, implements Cloneable {
    public void run() { /* do something */ }
    public Object clone() { /* make a copy */ }
}
```

Answer: CD
Section: All

Explanation/Reference:

QUESTION 57
Given:

```
class Foo {
    public int a = 3;
    public void addFive() { a += 5; System.out.print("f "); }
}
class Bar extends Foo {
    public int a = 8;
    public void addFive() { this.a += 5; System.out.print("b " ); }
}
```

Invoked with:

```
Foo f = new Bar();
f.addFive();
System.out.println(f.a);
```

What is the result?

A. `b 3`
B. `b 8`
C. `b 13`
D. `f 3`
E. `f 8`
F. `f 13`
G. Compilation fails.
H. An exception is thrown at runtime.

Answer: A
Section: All

Explanation/Reference:
`b 3`

QUESTION 58
Given:

```
import java.util.TreeSet;
public class Explorer2 {
    public static void main(String[] args) {
        TreeSet<Integer> s = new TreeSet<Integer>();
        TreeSet<Integer> subs = new TreeSet<Integer>();
        for(int i = 606; i < 613; i++)
            if(i%2 == 0) s.add(i);
        subs = (TreeSet)s.subSet(608, true, 611, true);
        s.add(629);
        System.out.println(s + " " + subs);
    }
}
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. `[608, 610, 612, 629] [608, 610]`
D. `[608, 610, 612, 629] [608, 610, 629]`
E. `[606, 608, 610, 612, 629] [608, 610]`

F. [606, 608, 610, 612, 629] [608, 610, 629]

Answer: E
Section: All

Explanation/Reference:
[606, 608, 610, 612, 629] [608, 610]

QUESTION 59
Given:

```
11. //insert code here
12.     private N min, max;
13.     public N getMin() { return min; }
14.     public N getMax() { return max; }
15.     public void add(N added) {
16.         if (min == null || added.doubleValue() < min.doubleValue())
17.             min = added;
18.         if (max == null || added.doubleValue() > max.doubleValue())
19.             max = added;
20.     }
21. }
```

Which two, inserted at line 11, will allow the code to compile? (Choose two.)

A. public class MinMax<?> {
B. public class MinMax<? extends Number> {
C. public class MinMax<N extends Object> {
D. public class MinMax<N extends Number> {
E. public class MinMax<? extends Object> {
F. public class MinMax<N extends Integer> {

Answer: DF
Section: All

Explanation/Reference:
A.
12 compilation errors

B.
```
Main.java:11: <identifier> expected
class MinMax<? extends Number> {
           ^
Main.java:11: '{' expected
class MinMax<? extends Number> {
                          ^
2 errors
```

C.
```
Main.java:16: cannot find symbol
symbol  : method doubleValue()
location: class java.lang.Object
if (min == null || added.doubleValue() < min.doubleValue())
                    ^
Main.java:16: cannot find symbol
symbol  : method doubleValue()
location: class java.lang.Object
if (min == null || added.doubleValue() < min.doubleValue())
                                       ^
```

```
Main.java:18: cannot find symbol
symbol  : method doubleValue()
location: class java.lang.Object
if (max == null || added.doubleValue() > max.doubleValue())
                    ^
Main.java:18: cannot find symbol
symbol  : method doubleValue()
location: class java.lang.Object
if (max == null || added.doubleValue() > max.doubleValue())
                                       ^
4 errors
```

D.
```
compiled successfully
```

E.
```
Main.java:1: <identifier> expected
class MinMax<? extends Object> {
           ^
Main.java:1: '{' expected
class MinMax<? extends Object> {
                         ^
2 errors
```

F.
```
compiled successfully
```

Exam B

QUESTION 1
A company that makes Computer Assisted Design (CAD) software has, within its application, some utility classes that are used to perform 3D rendering tasks. The company's chief scientist has just improved the performance of one of the utility classes' key rendering algorithms, and has assigned a programmer to replace the old algorithm with the new algorithm. When the programmer begins researching the utility classes, she is happy to discover that the algorithm to be replaced exists in only one class. The programmer reviews that class's API, and replaces the old algorithm with the new algorithm, being careful that her changes adhere strictly to the class's API. Once testing has begun, the programmer discovers that other classes that use the class she changed are no longer working properly. What design flaw is most likely the cause of these new bugs?

A. Inheritance
B. Tight coupling
C. Low cohesion
D. High cohesion
E. Loose coupling
F. Object immutability

Answer: B
Section: All

Explanation/Reference:

QUESTION 2
Given:

```
1. class ClassA {
2.     public int numberOfInstances;
```

```
3.     protected ClassA(int numberOfInstances) {
4.         this.numberOfInstances = numberOfInstances;
5.     }
6.  }
7.  public class ExtendedA extends ClassA {
8.     private ExtendedA(int numberOfInstances) {
9.         super(numberOfInstances);
10.    }
11.    public static void main(String[] args) {
12.        ExtendedA ext = new ExtendedA(420);
13.        System.out.print(ext.numberOfInstances);
14.    }
15. }
```

Which statement is true?

A. `420` is the output.
B. An exception is thrown at runtime.
C. All constructors must be declared `public`.
D. Constructors CANNOT use the `private` modifier.
E. Constructors CANNOT use the `protected` modifier.

Answer: A
Section: All

Explanation/Reference:
`420`

QUESTION 3
Given:

```
class ClassA {}
class ClassB extends ClassA {}
class ClassC extends ClassA {}
```

and:

```
ClassA p0 = new ClassA();
ClassB p1 = new ClassB();
ClassC p2 = new ClassC();
ClassA p3 = new ClassB();
ClassA p4 = new ClassC();
```

Which three are valid? (Choose three.)

A. `p0 = p1;`
B. `p1 = p2;`
C. `p2 = p4;`
D. `p2 = (ClassC)p1;`
E. `p1 = (ClassB)p3;`
F. `p2 = (ClassC)p4;`

Answer: AEF
Section: All

Explanation/Reference:
A.

```
compiled successfully
```

B.

```
incompatible types
found   : ClassC
required: ClassB
p1 = p2;
     ^
```

C.
```
incompatible types
found   : ClassA
required: ClassC
p2 = p4;
     ^
```

D.
```
inconvertible types
found   : ClassB
required: ClassC
p2 = (ClassC)p1;
           ^
```

E.
```
compiled successfully
```

F.
```
compiled successfully
```

QUESTION 4
Given:
```
class Thingy { Meter m = new Meter(); }
class Component { void go() { System.out.print("c"); } }
class Meter extends Component { void go() { System.out.print("m"); } } 8.
class DeluxeThingy extends Thingy {
   public static void main(String[] args) {
       DeluxeThingy dt = new DeluxeThingy();
       dt.m.go();
       Thingy t = new DeluxeThingy();
       t.m.go();
   }
}
```
Which two are true? (Choose two.)

A. The output is `mm`.
B. The output is `mc`.
C. `Component` is-a `Meter`.
D. `Component` has-a `Meter`.
E. `DeluxeThingy` is-a `Component`.
F. `DeluxeThingy` has-a `Component`.

Answer: AF
Section: All

Explanation/Reference:

QUESTION 5
Given:

```
10. interface Jumper { public void jump(); }
```

```
...
20. class Animal {}
...
30. class Dog extends Animal {
31. Tail tail;
32. }
...
40. class Beagle extends Dog implements Jumper{
41. public void jump() {}
42. }
...
50. class Cat implements Jumper{
51.    public void jump() {}
52. }
```

Which three are true? (Choose three.)

A. `Cat` is-a `Animal`
B. `Cat` is-a `Jumper`
C. `Dog` is-a `Animal`
D. `Dog` is-a `Jumper`
E. `Cat` has-a `Animal`
F. `Beagle` has-a `Tail`
G. `Beagle` has-a `Jumper`

Answer: BCF
Section: All

Explanation/Reference:

QUESTION 6
Given:

```
1. import java.util.*;
2. public class WrappedString {
3.    private String s;
4.    public WrappedString(String s) { this.s = s; }
5.    public static void main(String[] args) {
6.       HashSet<Object> hs = new HashSet<Object>();
7.       WrappedString ws1 = new WrappedString("aardvark");
8.       WrappedString ws2 = new WrappedString("aardvark");
9.       String s1 = new String("aardvark");
10.      String s2 = new String("aardvark");
11.      hs.add(ws1); hs.add(ws2); hs.add(s1); hs.add(s2);
12.      System.out.println(hs.size()); } }
```

What is the result?

A. `0`
B. `1`
C. `2`
D. `3`
E. `4`
F. Compilation fails.
G. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:
3

QUESTION 7
Given:

```
3. import java.util.*;
4. public class G1 {
5.    public void takeList(List<? extends String> list) {
6. //       insert code here
7.    }
8. }
```

Which three code fragments, inserted independently at line 6, will compile? (Choose three.)

A. `list.add("foo");`
B. `Object o = list;`
C. `String s = list.get(0);`
D. `list = new ArrayList<String>();`
E. `list = new ArrayList<Object>();`

Answer: BCD
Section: All

Explanation/Reference:
A.
```
Main.java:6: cannot find symbol
symbol  : method add(java.lang.String)
location: interface java.util.List<capture#824 of ? extends java.lang.String>
list.add("foo");
    ^
1 error
```

B.
```
compiled successfully
```

C.
```
compiled successfully
```

D.
```
compiled successfully
```

E.
```
Main.java:6: incompatible types
found   : java.util.ArrayList<java.lang.Object>
required: java.util.List<? extends java.lang.String>
list = new ArrayList<Object>();
       ^
1 error
```

QUESTION 8
Given that the elements of a `PriorityQueue` are ordered according to natural ordering, and:

```
import java.util.*;
```

```
public class GetInLine {
    public static void main(String[] args) {
        PriorityQueue<String> pq = new PriorityQueue<String>();
        pq.add("banana");
        pq.add("pear");
        pq.add("apple");
        System.out.println(pq.poll() + " " + pq.peek());
    }
}
```

What is the result?

A. `apple pear`
B. `banana pear`
C. `apple apple`
D. `apple banana`
E. `banana banana`

Answer: D
Section: All

Explanation/Reference:
`apple banana`

QUESTION 9
Given:

```
enum Example { ONE, TWO, THREE }
```

Which statement is true?

A. The expressions `(ONE == ONE)` and `ONE.equals(ONE)` are both guaranteed to be `true`.
B. The expression `(ONE < TWO)` is guaranteed to be `true` and `ONE.compareTo(TWO)` is guaranteed to be less than one.
C. The `Example` values cannot be used in a raw `java.util.HashMap`; instead, the programmer must use a `java.util.EnumMap`.
D. The `Example` values can be used in a `java.util.SortedSet`, but the set will NOT be sorted because enumerated types do NOT implement `java.lang.Comparable`.

Answer: A
Section: All

Explanation/Reference:

QUESTION 10
Given:

```
import java.util.*;
public class Mapit {
    public static void main(String[] args) {
        Set<Integer> set = new HashSet<Integer>();
        Integer i1 = 45;
        Integer i2 = 46;
        set.add(i1);
        set.add(i1);
```

```
        set.add(i2); System.out.print(set.size() + " ");
        set.remove(i1); System.out.print(set.size() + " ");
        i2 = 47;
        set.remove(i2); System.out.print(set.size() + " ");
    }
}
```

What is the result?

A. 2 1 0
B. 2 1 1
C. 3 2 1
D. 3 2 2
E. Compilation fails.
F. An exception is thrown at runtime.

Answer: B
Section: All

Explanation/Reference:
2 1 1

QUESTION 11
Given:

```
import java.util.*;
public class Explorer1 {
    public static void main(String[] args) {
        TreeSet<Integer> s = new TreeSet<Integer>();
        TreeSet<Integer> subs = new TreeSet<Integer>();
        for(int i = 606; i < 613; i++)
            if(i%2 == 0) s.add(i);
        subs = (TreeSet)s.subSet(608, true, 611, true);
        s.add(609);
        System.out.println(s + " " + subs);
    }
}
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. [608, 609, 610, 612] [608, 610]
D. [608, 609, 610, 612] [608, 609, 610]
E. [606, 608, 609, 610, 612] [608, 610]
F. [606, 608, 609, 610, 612] [608, 609, 610]

Answer: F
Section: All

Explanation/Reference:
[606, 608, 609, 610, 612] [608, 609, 610]

QUESTION 12
Given:

```
34. HashMap props = new HashMap();
```

```
35. props.put("key45", "some value");
36. props.put("key12", "some other value");
37. props.put("key39", "yet another value");
38. Set s = props.keySet();
39. //insert code here
```

What, inserted at line 39, will sort the keys in the props `HashMap`?

A. `Arrays.sort(s);`
B. `s = new TreeSet(s);`
C. `Collections.sort(s);`
D. `s = new SortedSet(s);`

Answer: B
Section: All

Explanation/Reference:

QUESTION 13
Which two statements are true? (Choose two.)

A. It is possible to synchronize `static` methods.
B. When a thread has yielded as a result of `yield()`, it releases its locks.
C. When a thread is sleeping as a result of `sleep()`, it releases its locks.
D. The `Object.wait()` method can be invoked only from a `synchronized` context.
E. The `Thread.sleep()` method can be invoked only from a `synchronized` context.
F. When the thread scheduler receives a `notify()` request, and notifies a thread, that thread immediately releases its lock.

Answer: AD
Section: All

Explanation/Reference:

QUESTION 14
Given:

```
public class TestOne implements Runnable {
    public static void main (String[] args) throws Exception {
        Thread t = new Thread(new TestOne());
        t.start();
        System.out.print("Started");
        t.join();
        System.out.print("Complete");
    }
    public void run() {
        for (int i = 0; i < 4; i++) {
            System.out.print(i);
        }
    }
}
```

What can be a result?

A. Compilation fails.
B. An exception is thrown at runtime.

C. The code executes and prints `StartedComplete`
D. The code executes and prints `StartedComplete0123`
E. The code executes and prints `Started0123Complete`

Answer: E
Section: All

Explanation/Reference:
`Started0123Complete`

QUESTION 15
Which three will compile and run without exception? (Choose three.)

A.
```
private synchronized Object o;
```
B.
```
void go() {
  synchronized() { /* code here */ }
```
C.
```
public synchronized void go() { /* code here */ }
```
D.
```
private synchronized(this) void go() { /* code here */ }
```
E.
```
void go() {
  synchronized(Object.class) { /* code here */ }
```
F.
```
void go() {
  Object o = new Object();
  synchronized(o) { /* code here */ }
```

Answer: CEF
Section: All

Explanation/Reference:

QUESTION 16
Given:
```
1. public class TestFive {
2.     private int x;
3.     public void foo() {
4.         int current = x;
5.         x = current + 1;
6.     }
7.     public void go() {
8.         for(int i = 0; i < 5; i++) {
9.             new Thread() {
10.                public void run() {
11.                    foo();
12.                    System.out.print(x + ", ");
13.                } }.start();
14.        } }
15. }
```
Which two changes, taken together, would guarantee the output: 1, 2, 3, 4, 5, ? (Choose two.)

A. move the line 12 print statement into the foo() method
B. change line 7 to public synchronized void go() {
C. change the variable declaration on line 2 to private volatile int x;
D. wrap the code inside the foo() method with a synchronized(this) block
E. wrap the for loop code inside the go() method with a synchronized block
synchronized(this) { // for loop code here }

Answer: AD
Section: All

Explanation/Reference:

```
public class TestFive {
    private int x;
    public void foo() {
        synchronized(this) {
            int current = x;
            x = current + 1;
            System.out.print(x + ", ");
        }
    }
    public void go() {
        for(int i = 0; i < 5; i++) {
            new Thread() {
                public void run() {
                    foo();
                } }.start();
        } }
}
```

QUESTION 17
Given that `t1` is a reference to a live thread, which is true?

A. The `Thread.sleep()` method can take `t1` as an argument.
B. The `Object.notify()` method can take `t1` as an argument.
C. The `Thread.yield()` method can take `t1` as an argument.
D. The `Thread.setPriority()` method can take `t1` as an argument.
E. The `Object.notify()` method arbitrarily chooses which thread to notify.

Answer: E
Section: All

Explanation/Reference:

QUESTION 18
Given:

```
Runnable r = new Runnable() {
    public void run() {
        System.out.print("Cat");
    }
};
Thread t = new Thread(r) {
    public void run() {
        System.out.print("Dog");
    }
};
t.start();
```

What is the result?

A. `Cat`
B. `Dog`

C. Compilation fails.
D. The code runs with no output.
E. An exception is thrown at runtime.

Answer: B
Section: All

Explanation/Reference:
Dog

QUESTION 19
Given:

```
1. public class Threads5 {
2.     public static void main (String[] args) {
3.         new Thread(new Runnable() {
4.             public void run() {
5.                 System.out.print("bar");
6.             }}).start();
7.     }
8. }
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. The code executes normally and prints `bar`.
D. The code executes normally, but nothing prints.

Answer: C
Section: All

Explanation/Reference:
bar

QUESTION 20
Given:

```
class One {
    void foo() { }
}
class Two extends One {
14. //    insert method here
}
```

Which three methods, inserted individually at line 14, will correctly complete class Two? (Choose three.)

A. int foo() { /* more code here */ }
B. void foo() { /* more code here */ }
C. public void foo() { /* more code here */ }
D. private void foo() { /* more code here */ }
E. protected void foo() { /* more code here */ }

Answer: BCE
Section: All

Explanation/Reference:
A.

```
Main.java:14: foo() in Two cannot override foo() in One; attempting to use incompatible return
type
found   : int
required: void
int foo() { /* more code here */ }
    ^
1 error
```

B.
```
compiled successfully
```

C.
```
compiled successfully
```

D.
```
Main.java:14: foo() in Two cannot override foo() in One; attempting to assign weaker access
privileges; was package
private void foo() { /* more code here */ }
           ^
1 error
```

E.
```
compiled successfully
```

QUESTION 21
Given:

```
abstract public class Employee {
   protected abstract double getSalesAmount();
   public double getCommision() {
      return getSalesAmount() * 0.15;
   }
}
class Sales extends Employee {
17. //    insert method here
}
```

Which two methods, inserted independently at line 17, correctly complete the `Sales` class? (Choose two.)

A. `double getSalesAmount() { return 1230.45; }`
B. `public double getSalesAmount() { return 1230.45; }`
C. `private double getSalesAmount() { return 1230.45; }`
D. `protected double getSalesAmount() { return 1230.45; }`

Answer: BD
Section: All

Explanation/Reference:
A.
```
Main.java:17: getSalesAmount() in Sales cannot override getSalesAmount() in Employee; attempting
to assign weaker access privileges; was protected
double getSalesAmount() { return 1230.45; }
       ^
1 error
```

B.
```
compiled successfully
```

C.
```
Main.java:17: getSalesAmount() in Sales cannot override getSalesAmount() in Employee; attempting
```

```
to assign weaker access privileges; was protected
private double getSalesAmount() { return 1230.45; }
           ^
1 error
```

D.
```
compiled successfully
```

QUESTION 22
Given:

```
1. class X {
2.     X() { System.out.print(1); }
3.     X(int x) {
4.         this(); System.out.print(2);
5.     }
6. }
7. public class Y extends X {
8.     Y() { super(6); System.out.print(3); }
9.     Y(int y) {
10.        this(); System.out.println(4);
11.    }
12.    public static void main(String[] a) { new Y(5); }
13.}
```

What is the result?

A. 13
B. 134
C. 1234
D. 2134
E. 2143
F. 4321

Answer: C
Section: All

Explanation/Reference:
1234

QUESTION 23
Given:

```
package com.sun.scjp;
public class Geodetics {
   public static final double DIAMETER = 12756.32; // kilometers
}
```

Which two correctly access the `DIAMETER` member of the `Geodetics` class? (Choose two.)

A.
```
import com.sun.scjp.Geodetics;
public class TerraCarta {
    public double halfway()
    { return Geodetics.DIAMETER/2.0; }
```

B.
```
import static com.sun.scjp.Geodetics;
public class TerraCarta{
public double halfway() { return DIAMETER/2.0; } }
```

C.
```
import static com.sun.scjp.Geodetics.*;
public class TerraCarta {
    public double halfway() { return DIAMETER/2.0; } }
```
D.
```
package com.sun.scjp;
public class TerraCarta {
    public double halfway() { return DIAMETER/2.0; } }
```

Answer: AC
Section: All

Explanation/Reference:

QUESTION 24
Given:

```
1. public class A {
2.     public void doit() {
3.     }
4.     public String doit() {
5.         return "a";
6.     }
7.     public double doit(int x) {
8.         return 1.0;
9.     }
10.}
```

What is the result?

A. An exception is thrown at runtime.
B. Compilation fails because of an error in line 7.
C. Compilation fails because of an error in line 4.
D. Compilation succeeds and no runtime errors with class `A` occur.

Answer: C
Section: All

Explanation/Reference:
```
Main.java:4: doit() is already defined in A
    public String doit() {
                  ^
1 error
```

QUESTION 25
Given:

```
35. String #name = "Jane Doe";
36. int $age = 24;
37. Double _height = 123.5;
38. double ~temp = 37.5;
```

Which two statements are true? (Choose two.)

A. Line 35 will not compile.
B. Line 36 will not compile.
C. Line 37 will not compile.

D. Line 38 will not compile.

Answer: AD
Section: All

Explanation/Reference:
A variable's name can be any legal identifier — an unlimited-length sequence of Unicode letters and digits, beginning with a letter, the dollar sign "$", or the underscore character "_".

QUESTION 26
Given:

```
1. public class ClassA {
2.     public void methodA() {
3.         ClassB classB = new ClassB();
4.         classB.getValue();
5.     }
6. }
7.
8. class ClassB {
9.     public ClassC classC;
10.    public String getValue() {
11.        return classC.getValue();
12.    }
13.}
14.
15.class ClassC {
16.    public String value;
17.    public String getValue() {
18.        value = "ClassC";
19.        return value;
20.    }
21.}
```

and:

```
ClassA a = new ClassA();
a.methodA();
```

What is the result?

A. Compilation fails.
B. `ClassC` is displayed.
C. The code runs with no output.
D. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:

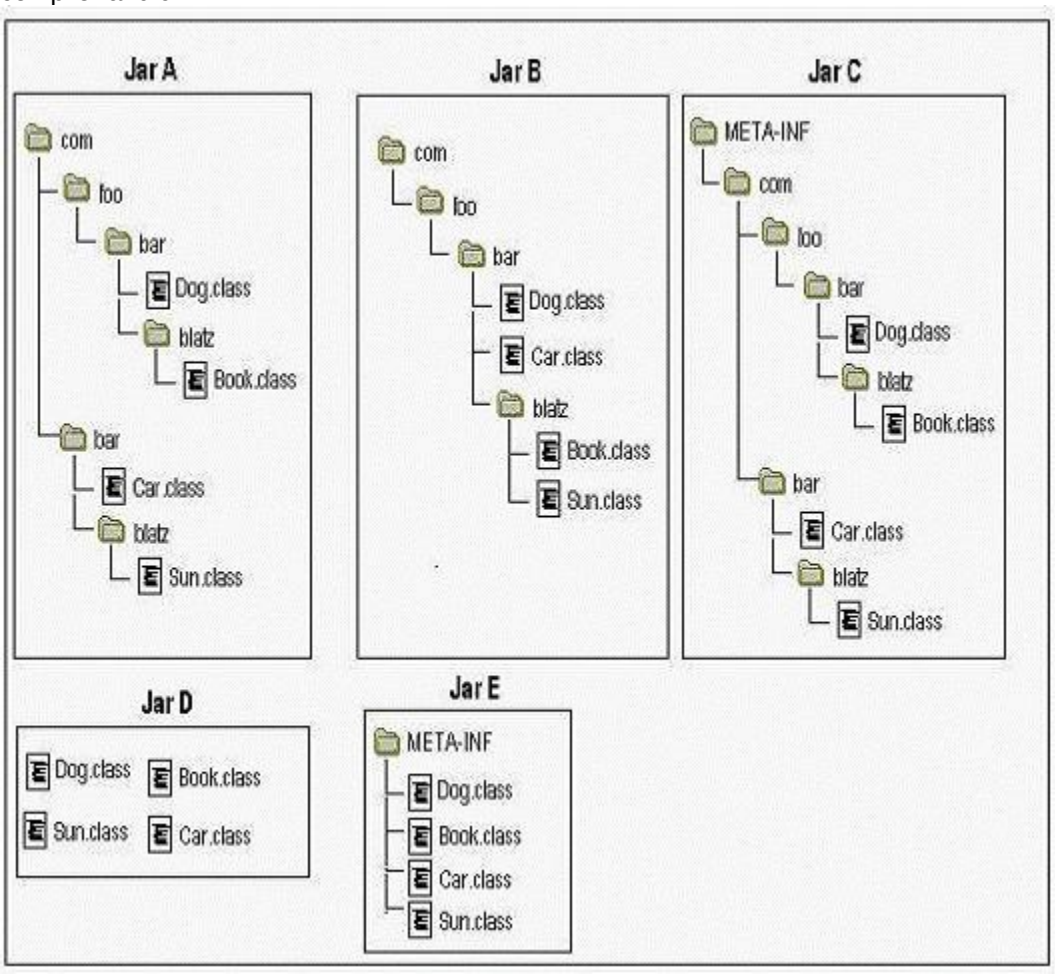
```
Exception in thread "main" java.lang.NullPointerException
    at ClassB.getValue(Main.java:11)
    at ClassA.methodA(Main.java:2)
    at a.methodA();
```

QUESTION 27

Click the Exhibit button.
Given the fully-qualified class names:

- com.foo.bar.Dog
- com.foo.bar.blatz.Book
- com.bar.Car
- com.bar.blatz.Sun

Which graph represents the correct directory structure for a JAR file from which those classes can be used by the compiler and JVM?



A. Jar A
B. Jar B
C. Jar C
D. Jar D
E. Jar E

Answer: A
Section: All

Explanation/Reference:

QUESTION 28
Given:

```
10. interface Foo { int bar(); }
```

```
11. public class Sprite {
12.     public int fubar( Foo foo ) { return foo.bar(); }
13.     public void testFoo() {
14.         fubar(
15.         //insert code here
16.         );
17.     }
18. }
```

Which code, inserted at line 15, allows the class `Sprite` to compile?

A. `Foo { public int bar() { return 1; }`
B. `new Foo { public int bar() { return 1; }`
C. `new Foo() { public int bar() { return 1; }`
D. `new class Foo { public int bar() { return 1; }`

Answer: C
Section: All

Explanation/Reference:

QUESTION 29
Given:

```
11. public enum Title {
12.     MR("Mr."), MRS("Mrs."), MS("Ms.");
13.     private final String title;
14.     private Title(String t) { title = t; }
15.     public String format(String last, String first) {
16.         return title + " " + first + " " + last;
17.     }
18. }
19.
20. public static void main(String[] args) {
21.     System.out.println(Title.MR.format("Doe", "John"));
22. }
```

What is the result?

A. `Mr. John Doe`
B. An exception is thrown at runtime.
C. Compilation fails because of an error in line 12.
D. Compilation fails because of an error in line 15.
E. Compilation fails because of an error in line 20.

Answer: A
Section: All

Explanation/Reference:
`Mr. John Doe`

QUESTION 30
Given the following six method names:

- `addListener`
- `addMouseListener`

- `setMouseListener`
- `deleteMouseListener`
- `removeMouseListener`
- `registerMouseListener`

How many of these method names follow JavaBean Listener naming rules?

A. 1
B. 2
C. 3
D. 4
E. 5

Answer: B
Section: All

Explanation/Reference:
Correct method names:
- `addMouseListener`
- `removeMouseListener`

Incorrect method names:
- `addListener` - missing listener's name
- `setMouseListener` - there are no set methods for listeners (only add and remove)
- `deleteMouseListener` - there are no delete methods for listeners (only add and remove)
- `registerMouseListener` - there are no register methods for listeners (only add and remove)

QUESTION 31
Given:

```
09. class Line {
10.     public static class Point {}
11. }
12.
13. class Triangle {
14.     public Triangle(){
15.     //  insert code here
16.     }
17. }
```

Which code, inserted at line 15, creates an instance of the `Point` class defined in `Line`?

A. `Point p = new Point();`
B. `Line.Point p = new Line.Point();`
C. The `Point` class cannot be instatiated at line 15.
D. `Line l = new Line() ; l.Point p = new l.Point();`

Answer: B
Section: All

Explanation/Reference:
A.

```
Main.java:15: cannot find symbol
symbol  : class Point
location: class Triangle
    Point p = new Point();
    ^
Main.java:15: cannot find symbol
```

```
symbol  : class Point
location: class Triangle
    Point p = new Point();
                  ^
2 errors
```

B.

```
compiled successfully
```

C.

```
false
```

D.

```
Main.java:15: package l does not exist
    Line l = new Line() ; l.Point p = new l.Point();
                        ^
Main.java:15: package l does not exist
    Line l = new Line() ; l.Point p = new l.Point();
                                        ^
2 errors
```

QUESTION 32
Given

```
11. public interface Status {
12.    /* insert code here */ int MY_VALUE = 10;
13. }
```

Which three are valid on line 12? (Choose three.)

A. final
B. static
C. native
D. public
E. private
F. abstract
G. protected

Answer: ABD
Section: All

Explanation/Reference:

QUESTION 33
Given this code from Class B:

```
25. A a1 = new A();
26. A a2 = new A();
27. A a3 = new A();
28. System.out.println(A.getInstanceCount());
```

What is the result?

```
1. public class A{
2.
3.     private int counter = 0;
```

```
4.
5.     public static int getInstanceCount() {
6.         return counter;
7.     }
8.
9.     public A() {
10.        counter++;
11.    }
12.
13. }
```

A. Compilation of class `A` fails.
B. Line 28 prints the value 3 to System.out.
C. Line 28 prints the value 1 to System.out.
D. A runtime error occurs when line 25 executes.
E. Compilation fails because of an error on line 28.

Answer: A
Section: All

Explanation/Reference:
`counter` variable is being used inside a `static` context.

QUESTION 34
Given classes defined in two different files:

```
1. package util;
2. public class BitUtils {
3.     public static void process(byte[] b) { /* more code here */ }
4. }
```

```
1. package app;
2. public class SomeApp {
3.     public static void main(String[] args) {
4.         byte[] bytes = new byte[256];
5. //         insert code here
6.     }
7. }
```

What is required at line 5 in class `SomeApp` to use the process method of `BitUtils`?

A. `process(bytes);`
B. `BitUtils.process(bytes);`
C. `util.BitUtils.process(bytes);`
D. `SomeApp` cannot use methods in `BitUtils`.
E. `import util.BitUtils.*; process(bytes);`

Answer: C
Section: All

Explanation/Reference:

QUESTION 35

Which three code fragments, added individually at line 29, produce the output 100? (Choose three.)

```
10. class Inner {
11.    private int x;
12.    public void setX( int x ){ this.x = x; }
13.    public int getX(){ return x;}
14. }
15.
16. class Outer {
17.    private Inner y;
18.    public void setY( Inner y ){ this.y = y; }
19.    public Inner getY() { return y; }
20. }
21.
22. public class Gamma {
23.    public static void main(String[] args) {
24.       Outer o = new Outer();
25.       Inner i = new Inner();
26.       int n = 10;
27.       i.setX(n);
28.       o.setY(i);
29.       // insert code here 29
30.       System.out.println(o.getY().getX());
31.
32.    }
33. }
```

A. `n = 100;`
B. `i.setX( 100 );`
C. `o.getY().setX( 100 );`
D. `i = new Inner(); i.setX( 100 );`
E. `o.setY( i ); i = new Inner(); i.setX( 100 );`
F. `i = new Inner(); i.setX( 100 ); o.setY( i );`

Answer: BCF
Section: All

Explanation/Reference:
A. 10

B. 100

C. 100

D. 10

E. 10

F. 100

QUESTION 36
Given:

```
class Snoochy {
   Boochy booch;

   public Snoochy() { booch = new Boochy(this); }
}
```

```
class Boochy {
   Snoochy snooch;
   public Boochy(Snoochy s) { snooch = s; }
}
```

And the statements:

```
21. public static void main(String[] args) {
22.     Snoochy snoog = new Snoochy();
23.     snoog = null;
24.     //    more code here
25. }
```

Which statement is true about the objects referenced by `snoog`, `snooch`, and `booch` immediately after line 23 executes?

A. None of these objects are eligible for garbage collection.
B. Only the object referenced by `booch` is eligible for garbage collection.
C. Only the object referenced by `snoog` is eligible for garbage collection.
D. Only the object referenced by `snooch` is eligible for garbage collection.
E. The objects referenced by `snooch` and `booch` are eligible for garbage collection.

Answer: E
Section: All

Explanation/Reference:

QUESTION 37
Given:

```
03. interface Animal { void makeNoise(); }
04. class Horse implements Animal {
05.     Long weight = 1200L;
06.     public void makeNoise() { System.out.println("whinny"); }
07. }
08. public class Icelandic extends Horse {
09.     public void makeNoise() { System.out.println("vinny"); }
10.     public static void main(String[] args) {
11.         Icelandic i1 = new Icelandic();
12.         Icelandic i2 = new Icelandic();
13.         Icelandic i3 = new Icelandic();
14.         i3 = i1; i1 = i2; i2 = null; i3 = i1;
15.     }
16. }
```

When line 15 is reached, how many objects are eligible for the garbage collector?

A. 0
B. 1
C. 2
D. 3
E. 4
F. 6

Answer: E
Section: All

Explanation/Reference:

QUESTION 38
Given:

```
01. public static void test(String str) {
02.     int check = 4;
03.     if (check = str.length()) {
04.         System.out.print(str.charAt(check -= 1) +", ");
05.     } else {
06.         System.out.print(str.charAt(0) + ", ");
07.     }
08. }
```

and the invocation:

```
test("four");
test("tee");
test("to");
```

What is the result?

A. r, t, t,
B. r, e, o,
C. Compilation fails.
D. An exception is thrown at runtime.

Answer: C
Section: All

Explanation/Reference:

```
Main.java:3: incompatible types
found   : int
required: boolean
    if (check = str.length()) {
           ^
1 error
```

QUESTION 39
Given classes defined in two different files:

```
1. package util;
2. public class BitUtils {
3.     private static void process(byte[] b) {}
4. }
```

and

```
1. package app;
2. public class SomeApp {
3.     public static void main(String[] args) {
4.         byte[] bytes = new byte[256];
5.         // insert code here
6.     }
7. }
```

What is required at line 5 in class `SomeApp` to use the `process` method of `BitUtils`?

A. `process(bytes);`
B. `BitUtils.process(bytes);`
C. `app.BitUtils.process(bytes);`
D. `util.BitUtils.process(bytes);`
E. `import util.BitUtils.*; process(bytes);`
F. `SomeApp` cannot use the `process` method in `BitUtils`.

Answer: F
Section: All

Explanation/Reference:
`util.BitUtils.process` method is declared `private` and cannot be used outside `BitUtils` class.

QUESTION 40
Given:

```
1. public class Pass2 {
2.     public void main(String [] args) {
3.         int x = 6;
4.         Pass2 p = new Pass2();
5.         p.doStuff(x);
6.         System.out.print(" main x = " + x);
7.     }
8.
9.     void doStuff(int x) {
10.        System.out.print(" doStuff x = " + x++);
11.    }
12. }
```

And the command-line invocations:

```
javac Pass2.java
java Pass2 5
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. `doStuff x = 6 main x = 6`
D. `doStuff x = 6 main x = 7`
E. `doStuff x = 7 main x = 6`
F. `doStuff x = 7 main x = 7`

Answer: B
Section: All

Explanation/Reference:
An exception occurs because it's missing static at line 2 (`public static void main(String [] args) {}`

QUESTION 41
Given:

```
public class Test {
```

```
    public enum Dogs {collie, harrier};
    public static void main(String [] args) {
        Dogs myDog = Dogs.collie;
        switch (myDog) {
        case collie:
            System.out.print("collie ");
        case harrier:
            System.out.print("harrier ");
        }
    }
}
```

What is the result?

A. `collie`
B. `harrier`
C. Compilation fails.
D. `collie harrier`
E. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:
`collie harrier`

There are no `break;`'s between `case`'s ("fall-through")

QUESTION 42
Given:

```
public class Donkey {
    public static void main(String[] args) {
        boolean assertsOn = false;
        assert (assertsOn) : assertsOn = true;
        if(assertsOn) {
            System.out.println("assert is on");
        }
    }
}
```

If class `Donkey` is invoked twice, the first time without assertions enabled, and the second time with assertions enabled, what are the results?

A. no output
B. no output
   `assert is on`
C. `assert is on`
D. no output
   An AssertionError is thrown.
E. `assert is on`
   An AssertionError is thrown.

Answer: D
Section: All

Explanation/Reference:

QUESTION 43
Given:

```
01. static void test() throws Error {
02.     if (true) throw new AssertionError();
03.     System.out.print("test ");
04. }
05. public static void main(String[] args) {
06.     try { test(); }
07.     catch (Exception ex) { System.out.print("exception "); }
08.     System.out.print("end ");
09. }
10. }
```

What is the result?

A. `end`
B. Compilation fails.
C. `exception end`
D. `exception test end`
E. A Throwable is thrown by main.
F. An Exception is thrown by main.

Answer: E
Section: All

Explanation/Reference:

```
Exception in thread "main" java.lang.AssertionError
    at A.test(Main.java:2)
    at A.main(Main.java:6)
```

QUESTION 44
Given:

```
1. class TestException extends Exception { }
2. class A {
3.     public String sayHello(String name) throws TestException {
4.         if(name == null) throw new TestException();
5.         return "Hello " + name;
6.     }
7. }
8. public class TestA {
9.     public static void main(String[] args) {
10.        new A().sayHello("Aiko");
11.    }
12.}
```

Which statement is true?

A. Compilation succeeds.
B. Class `A` does not compile.
C. The method declared on line 9 cannot be modified to throw `TestException`.
D. `TestA` compiles if line 10 is enclosed in a `try`/`catch` block that catches `TestException`.

Answer: D
Section: All

Explanation/Reference:
```
Main.java:10: unreported exception TestException; must be caught or declared to be thrown
       new A().sayHello("Aiko");
                        ^
1 error
```

QUESTION 45
Given:

```
static class A {
   void process() throws Exception { throw new Exception(); }
}

static class B extends A {
   void process() { System.out.println("B"); }
}

public static void main(String[] args) {
   new B().process();
}
```

What is the result?

A. B
B. The code runs with no output.
C. Compilation fails because of an error in line 12.
D. Compilation fails because of an error in line 15.
E. Compilation fails because of an error in line 18.

Answer: A
Section: All

Explanation/Reference:

QUESTION 46
Given:

```
public class Foo {
   static int[] a;
   static { a[0]=2; }
   public static void main( String[] args ) {}
}
```

Which exception or error will be thrown when a programmer attempts to run this code?

A. java.lang.StackOverflowError
B. java.lang.IllegalStateException
C. java.lang.ExceptionInInitializerError
D. java.lang.ArrayIndexOutOfBoundsException

Answer: C
Section: All

Explanation/Reference:

```
Exception in thread "main" java.lang.ExceptionInInitializerError
Caused by: java.lang.NullPointerException
    at Foo.<clinit>(Main.java:3)
Could not find the main class: Foo. Program will exit.
```

QUESTION 47
Given:

```
11. public static void main(String[] args) {
12.    Integer i = new Integer(1) + new Integer(2);
13.    switch(i) {
14.    case 3: System.out.println("three"); break;
15.    default: System.out.println("other"); break;
16.    }
17. }
```

What is the result?

A. `three`
B. `other`
C. An exception is thrown at runtime.
D. Compilation fails because of an error on line 12.
E. Compilation fails because of an error on line 13.
F. Compilation fails because of an error on line 15.

Answer: A
Section: All

Explanation/Reference:

```
three
```

QUESTION 48
Given:

```
1. public class TestString3 {
2.     public static void main(String[] args) {
3. //        insert code here
4.
5.         System.out.println(s);
6.     }
7. }
```

Which two code fragments, inserted independently at line `3`, generate the output `4247`? (Choose two.)

A.
```
String s = "123456789";
s = (s-"123").replace(1,3,"24") - "89";
```
B.
```
StringBuffer s = new StringBuffer("123456789");
s.delete(0,3).replace(1,3,"24").delete(4,6);
```
C.
```
StringBuffer s = new StringBuffer("123456789");
s.substring(3,6).delete(1,3).insert(1, "24");
```
D.
```
StringBuilder s = new StringBuilder("123456789");
s.substring(3,6).delete(1,2).insert(1, "24");
```
E.
```
StringBuilder s = new StringBuilder("123456789");
s.delete(0,3).delete(1,3).delete(2,5).insert(1, "24");
```

Answer: BE
Section: All

Explanation/Reference:
A.
```
Main.java:4: operator - cannot be applied to java.lang.String,java.lang.String
s = (s-"123").replace(1,3,"24") - "89";
     ^
1 error
```

B. 4247

C.
```
Main.java:4: cannot find symbol
symbol  : method delete(int,int)
location: class java.lang.String
s.substring(3,6).delete(1,3).insert(1, "24");
            ^
1 error
```

D.
```
Main.java:4: cannot find symbol
symbol  : method delete(int,int)
location: class java.lang.String
s.substring(3,6).delete(1,2).insert(1, "24");
            ^
1 error
```

E. 4247

QUESTION 49
Given:

1. d is a valid, non-null Date object
2. df is a valid, non-null DateFormat object set to the current locale

What outputs the current locale's country name and the appropriate version of d's date?

A.
```
Locale loc = Locale.getLocale();
System.out.println(loc.getDisplayCountry()
       + " " + df.format(d));
```
B.
```
Locale loc = Locale.getDefault();
System.out.println(loc.getDisplayCountry()
       + " " + df.format(d));
```
C.
```
Locale loc = Locale.getLocale();
System.out.println(loc.getDisplayCountry()
       + " " + df.setDateFormat(d));
```
D.
```
Locale loc = Locale.getDefault();
System.out.println(loc.getDisplayCountry()
       + " " + df.setDateFormat(d));
```

Answer: B
Section: All

Explanation/Reference:

QUESTION 50
Given:

```
18. import java.util.Date;
19. import java.text.DateFormat;
20.
21. DateFormat df;
22. Date date = new Date();
23. //insert code here
24. String s = df.format(date);
```

Which code fragment, inserted at line 23, allows the code to compile?

A. `df = new DateFormat();`
B. `df = Date.getFormat();`
C. `df = date.getFormat();`
D. `df = DateFormat.getFormat();`
E. `df = DateFormat.getInstance();`

Answer: E
Section: All

Explanation/Reference:
A.

```
Main.java:23: java.text.DateFormat is abstract; cannot be instantiated
df = new DateFormat();
     ^
1 error
```

B.

```
Main.java:23: cannot find symbol
symbol  : method getFormat()
location: class java.util.Date
df = Date.getFormat();
        ^
1 error
```

C.

```
Main.java:23: cannot find symbol
symbol  : method getFormat()
location: class java.util.Date
df = date.getFormat();
        ^
1 error
```

D.

```
Main.java:23: cannot find symbol
symbol  : method getFormat()
location: class java.text.DateFormat
df = DateFormat.getFormat();
             ^
1 error
```

E.

```
compiled successfully
```

QUESTION 51
Given:

```
1. public class BuildStuff {
2.     public static void main(String[] args) {
3.         Boolean test = new Boolean(true);
4.         Integer x = 343;
5.         Integer y = new BuildStuff().go(test, x);
6.         System.out.println(y);
7.     }
8.     int go(Boolean b, int i) {
9.         if(b) return (i/7);
10.        return (i/49);
11.    }
12. }
```

What is the result?

A. 7
B. 49
C. 343
D. Compilation fails.
E. An exception is thrown at runtime.

Answer: B
Section: All

Explanation/Reference:
49

QUESTION 52
Given:

```
import java.io.*;
public class Forest implements Serializable {
    private Tree tree = new Tree();
    public static void main(String [] args) {
        Forest f = new Forest();
        try {

            FileOutputStream fs = new FileOutputStream("Forest.ser");
            ObjectOutputStream os = new ObjectOutputStream(fs);
            os.writeObject(f); os.close();
        } catch (Exception ex) { ex.printStackTrace(); }
    }
}

class Tree { }
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. An instance of `Forest` is serialized.
D. An instance of `Forest` and an instance of `Tree` are both serialized.

Answer: B
Section: All

Explanation/Reference:

```
java.io.FileNotFoundException: Forest.ser (Permission denied)
    at java.io.FileOutputStream.open(Native Method)
    at java.io.FileOutputStream.<init>(FileOutputStream.java:194)
    at java.io.FileOutputStream.<init>(FileOutputStream.java:84)
    at Forest.main(Main.java:8)
```

QUESTION 53
Given:

```
04. import java.io.*;
05.
06. public class Talk {
07.    public static void main(String[] args) {
08.       Console c = new Console();
09.       String pw;
10.       System.out.print("password: ");
11.       pw = c.readLine();
12.       System.out.println("got " + pw);
13.    }
14.}
```

If the user types the password `aiko` when prompted, what is the result?

A.
```
password:
got
```
B.
```
password:
got aiko
```
C.
```
password: aiko
got aiko
```
D. An exception is thrown at runtime.
E. Compilation fails due to an error on line 8.

Answer: E
Section: All

Explanation/Reference:

```
Main.java:8: Console() has private access in java.io.Console
      Console c = new Console();
                  ^
1 error
```

Line 8 should be:

```
Console c = System.console();
```

QUESTION 54
Given:

```
1. public class LineUp {
2.    public static void main(String[] args) {
3.       double d = 12.345;
4.       // insert code here
5.    }
6. }
```

Which code fragment, inserted at line 4, produces the output `| 12.345|`?

A. `System.out.printf("|%7d| \n", d);`
B. `System.out.printf("|%7f| \n", d);`
C. `System.out.printf("|%3.7d| \n", d);`
D. `System.out.printf("|%3.7f| \n", d);`
E. `System.out.printf("|%7.3d| \n", d);`
F. `System.out.printf("|%7.3f| \n", d);`

Answer: F
Section: All

Explanation/Reference:
A. `|`

```
Exception in thread "main" java.util.IllegalFormatConversionException: d != java.lang.Double
    at java.util.Formatter$FormatSpecifier.failConversion(Formatter.java:3999)
    at java.util.Formatter$FormatSpecifier.printInteger(Formatter.java:2709)
    at java.util.Formatter$FormatSpecifier.print(Formatter.java:2661)
    at java.util.Formatter.format(Formatter.java:2433)
    at java.io.PrintStream.format(PrintStream.java:920)
    at java.io.PrintStream.printf(PrintStream.java:821)
    at LineUp.main(Main.java:4)
```

B. `|12.345000|`

C.

```
Exception in thread "main" java.util.IllegalFormatPrecisionException: 7
    at java.util.Formatter$FormatSpecifier.checkInteger(Formatter.java:2892)
    at java.util.Formatter$FormatSpecifier.<init>(Formatter.java:2643)
    at java.util.Formatter.parse(Formatter.java:2480)
    at java.util.Formatter.format(Formatter.java:2414)
    at java.io.PrintStream.format(PrintStream.java:920)
    at java.io.PrintStream.printf(PrintStream.java:821)
    at LineUp.main(Main.java:4)
```

D. `|12.3450000|`

E.

```
Exception in thread "main" java.util.IllegalFormatPrecisionException: 3
    at java.util.Formatter$FormatSpecifier.checkInteger(Formatter.java:2892)
    at java.util.Formatter$FormatSpecifier.<init>(Formatter.java:2643)
    at java.util.Formatter.parse(Formatter.java:2480)
    at java.util.Formatter.format(Formatter.java:2414)
    at java.io.PrintStream.format(PrintStream.java:920)
    at java.io.PrintStream.printf(PrintStream.java:821)
    at LineUp.main(Main.java:4)
```

F. `| 12.345|`

QUESTION 55
Given:

```
1. public class Threads4 {
2.     public static void main (String[] args) {
3.         new Threads4().go();
4.     }
5.     public void go() {
6.         Runnable r = new Runnable() {
7.             public void run() {
8.                 System.out.print("foo");
```

```
9.            }
10.       };
11.       Thread t = new Thread(r);
12.       t.start();
13.       t.start();
14.   }
15.}
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. The code executes normally and prints `foo`.
D. The code executes normally, but nothing is printed.

Answer: B
Section: All

Explanation/Reference:

```
 foo
 Exception in thread "main" java.lang.IllegalThreadStateException
    at java.lang.Thread.start(Thread.java:638)
    at Threads4.go(Main.java:13)
    at Threads4.main(Main.java:3)
```

QUESTION 56
Given:

```
1. public class Mud {
2.     //insert code here
3.        System.out.println("hi");
4.     }
5. }
```

And the following five fragments:

```
public static void main(String...a) {
public static void main(String.* a) {
public static void main(String... a) {
public static void main(String[]... a) {
public static void main(String...[] a) {
```

How many of the code fragments, inserted independently at line 2, compile?

A. 0
B. 1
C. 2
D. 3
E. 4
F. 5

Answer: D
Section: All

Explanation/Reference:

```
•public static void main(String...a) {
compiled successfully
```

```
•public static void main(String.* a) {
Main.java:2: <identifier> expected
   public static void main(String.* a) {
                               ^
Main.java:2: ')' expected
   public static void main(String.* a) {
                                ^
Main.java:2: ';' expected
   public static void main(String.* a) {
                                  ^
3 errors

•public static void main(String... a) {
compiled successfully

•public static void main(String[]... a) {
compiled successfully

•public static void main(String...[] a) {
Main.java:2: <identifier> expected
   public static void main(String...[] a) {
                                  ^
Main.java:2: ';' expected
   public static void main(String...[] a) {
                                   ^
Main.java:2: <identifier> expected
   public static void main(String...[] a) {
                                     ^
3 errors
```

QUESTION 57
Given:

```
static void test() throws RuntimeException {
    try {
        System.out.print("test ");
        throw new RuntimeException();
    }
    catch (Exception ex) { System.out.print("exception "); }
}
public static void main(String[] args) {
    try { test(); }
    catch (RuntimeException ex) { System.out.print("runtime "); }
    System.out.print("end ");
}
```

What is the result?

A. `test end`
B. Compilation fails.
C. `test runtime end`
D. `test exception end`
E. A Throwable is thrown by main at runtime.

Answer: D
Section: All

Explanation/Reference:
`test exception end`

QUESTION 58
Given:

```
11. public class Test {
12.    public static void main(String [] args) {
13.       int x = 5;
14.       boolean b1 = true;
15.       boolean b2 = false;
16.
17.       if ((x == 4) && !b2 )
18.          System.out.print("1 ");
19.       System.out.print("2 ");
20.       if ((b2 = true) && b1 )
21.          System.out.print("3 ");
22.    }
23. }
```

What is the result?

A. 2
B. 3
C. 1 2
D. 2 3
E. 1 2 3
F. Compilation fails.
G. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:
2 3

QUESTION 59
Given:

```
int x = 0;
int y = 10;
do {
   y--;
   ++x;
} while (x < 5);
System.out.print(x + "," + y);
```

What is the result?

A. 5,6
B. 5,5
C. 6,5
D. 6,6

Answer: B
Section: All

Explanation/Reference:
5,5

QUESTION 60
Given:

```
3. import java.util.*;
4. public class Hancock {
5.         // insert code here
6.         list.add("foo");
7.     }
8. }
```

Which two code fragments, inserted independently at line 5, will compile without warnings? (Choose two.)

A. `public void addStrings(List list) {`
B. `public void addStrings(List<String> list) {`
C. `public void addStrings(List<? super String> list) {`
D. `public void addStrings(List<? extends String> list) {`

Answer: BC
Section: All

Explanation/Reference:

Exam C

QUESTION 1
Given:

```
11. String test = "Test A. Test B. Test C.";
12. // insert code here
13. String[] result = test.split(regex);
```

Which regular expression, inserted at line 12, correctly splits test into "Test A", "Test B", and "Test C"?

A. `String regex = "";`
B. `String regex = " ";`
C. `String regex = ".*";`
D. `String regex = "\\s";`
E. `String regex = "\\.\\s*";`
F. `String regex = "\\w[ \.] +";`

Answer: E
Section: All

Explanation/Reference:
A.

```
T
e
s
t

A
.

T
e
s
```

```
t

B
.

T
e
s
t

C
.
```

B.
```
Test
A.
Test
B.
Test
C.
```

C.
```
<no output>
```

D.
```
Test
A.
Test
B.
Test
C.
```

E.
```
Test A
Test B
Test C
```

F.
```
Main.java:12: illegal escape character
    String regex = "\\w[ \.] +";
                       ^
1 error
```

QUESTION 2
Given:

```
1. interface A { public void aMethod(); }
2. interface B { public void bMethod(); }
3. interface C extends A,B { public void cMethod(); }
4. class D implements B {
5.     public void bMethod(){}
6. }
7. class E extends D implements C {
8.     public void aMethod(){}
9.     public void bMethod(){}
10.    public void cMethod(){}
11.}
```

What is the result?

A. Compilation fails because of an error in line 3.
B. Compilation fails because of an error in line 7.
C. Compilation fails because of an error in line 9.
D. If you define `D e = new E()`, then `e.bMethod()` invokes the version of `bMethod()` defined in Line 5.
E. If you define `D e = (D)(new E())`, then `e.bMethod()` invokes the version of `bMethod()` defined in Line 5.
F. If you define `D e = (D)(new E())`, then `e.bMethod()` invokes the version of `bMethod()` defined in Line 9.

Answer: F
Section: All

Explanation/Reference:

QUESTION 3
What is the result?

```
1. public class SimpleCalc {
2.     public int value;
3.     public void calculate() { value += 7; }
4. }
```

and:

```
1. public class MultiCalc extends SimpleCalc {
2.     public void calculate() { value -= 3; }
3.     public void calculate(int multiplier) {
4.         calculate();
5.         super.calculate();
6.         value *= multiplier;
7.     }
8.     public static void main(String[] args) {
9.         MultiCalc calculator = new MultiCalc();
10.        calculator.calculate(2);
11.        System.out.println("Value is: " + calculator.value);
12.    }
13.}
```

A. `Value is: 8`
B. Compilation fails.
C. `Value is: 12`
D. `Value is: -12`
E. The code runs with no output.
F. An exception is thrown at runtime.

Answer: A
Section: All

Explanation/Reference:
`Value is: 8`

QUESTION 4

Given:

```
1. class Mammal {
2. }
3.
4. class Raccoon extends Mammal {
5.     Mammal m = new Mammal();
6. }
7.
8. class BabyRaccoon extends Mammal {
9. }
```

Which four statements are true? (Choose four.)

A. `Raccoon` is-a `Mammal`.
B. `Raccoon` has-a `Mammal`.
C. `BabyRaccoon` is-a `Mammal`.
D. `BabyRaccoon` is-a `Raccoon`.
E. `BabyRaccoon` has-a `Mammal`.
F. `BabyRaccoon` is-a `BabyRaccoon`.

Answer: ABCF
Section: All

Explanation/Reference:

QUESTION 5
Given:

```
10. interface A { void x(); }
11. class B implements A { public void x() {} public void y() {} }
12. class C extends B { public void x() {} }
```

And:

```
20. java.util.List<A> list = new java.util.ArrayList<A>();
21. list.add(new B());
22. list.add(new C());
23. for (A a : list) {
24.     a.x();
25.     a.y();
26. }
```

What is the result?

A. The code runs with no output.
B. An exception is thrown at runtime.
C. Compilation fails because of an error in line 20.
D. Compilation fails because of an error in line 21.
E. Compilation fails because of an error in line 23.
F. Compilation fails because of an error in line 25.

Answer: F
Section: All

Explanation/Reference:

```
Main.java:25: cannot find symbol
symbol  : method y()
location: interface A
    a.y();
     ^
1 error
```

QUESTION 6
Given:

```
1.
2. public class Hi {
3.    void m1() { }
4.    protected void m2() { }
5. }
6. class Lois extends Hi {
7. //    insert code here
8. }
```

Which four code fragments, inserted independently at line 7, will compile? (Choose four.)

A. `public void m1() { }`
B. `protected void m1() { }`
C. `private void m1() { }`
D. `void m2() { }`
E. `public void m2() { }`
F. `protected void m2() { }`
G. `private void m2() { }`

Answer: ABEF
Section: All

Explanation/Reference:
A.
```
compiled successfully
```

B.
```
compiled successfully
```

C.
```
Main.java:7: m1() in Lois cannot override m1() in Hi; attempting to assign weaker access privileges;
was package
private void m1() { }
           ^
1 error
```

D.
```
Main.java:7: m2() in Lois cannot override m2() in Hi; attempting to assign weaker access privileges;
was protected
void m2() { }
     ^
1 error
```

E.
```
compiled successfully
```

F.
```
compiled successfully
```

G.

```
Main.java:7: m2() in Lois cannot override m2() in Hi; attempting to assign weaker access privileges;
was protected
private void m2() { }
           ^
1 error
```

QUESTION 7
Which four statements are true? (Choose four.)

A. Has-a relationships should never be encapsulated.
B. Has-a relationships should be implemented using inheritance.
C. Has-a relationships can be implemented using instance variables.
D. Is-a relationships can be implemented using the `extends` keyword.
E. Is-a relationships can be implemented using the `implements` keyword.
F. The relationship between Movie and Actress is an example of an is-a relationship.
G. An array or a collection can be used to implement a one-to-many has-a relationship.

Answer: CDEG
Section: All

Explanation/Reference:

QUESTION 8
Given:

```
01. public class Hello {
02.    String title;
03.    int value;
04.
05.     public Hello() {
06.         title += " World";
07.     }
08.
09.     public Hello(int value) {
10.         this.value = value;
11.         title = "Hello";
12.         Hello();
13.     }
14. }
```

and:

```
Hello c = new Hello(5);
System.out.println(c.title);
```

What is the result?

A. `Hello`
B. `Hello World`
C. Compilation fails.
D. `Hello World 5`
E. The code runs with no output.
F. An exception is thrown at runtime.

Answer: C

Section: All

Explanation/Reference:

```
Main.java:12: cannot find symbol
symbol  : method Hello()
location: class Hello
       Hello();
       ^
1 error
```

QUESTION 9
Given:

```
1. package geometry;
2.
3. public class Hypotenuse {
4.     public InnerTriangle it = new InnerTriangle();
5.
6.     class InnerTriangle {
7.         public int base;
8.         public int height;
9.     }
10. }
```

Which statement is true about the class of an object that can reference the variable `base`?

A. It can be any class.
B. No class has access to base.
C. The class must belong to the `geometry` package.
D. The class must be a subclass of the class `Hypotenuse`.

Answer: C
Section: All

Explanation/Reference:

QUESTION 10
Click the Exhibit button.

```
1. public class A {
2.
3.     private int counter = 0;
4.
5.     public static int getInstanceCount(){
6.         return counter;
7.     }
8.
9.     public A(){
10.        counter++;
11.    }
12. }
13.
```

Given this code from Class `B`:

```
A a1 = new A();
A a2 = new A();
A a3 = new A();
System.out.println(A.getInstanceCount());
```

What is the result?

A. Compilation of class `A` fails.
B. Line 28 prints the value 3 to `System.out`.
C. Line 28 prints the value 1 to `System.out`.
D. A runtime error occurs when line 25 executes.
E. Compilation fails because of an error on line 28.

Answer: A
Section: All

Explanation/Reference:

```
Main.java:6: non-static variable counter cannot be referenced from a static context
       return counter;
              ^
1 error
```

QUESTION 11
Given:

```
10. interface Data { public void load(); }
11. abstract class Info { public abstract void load(); }
```

Which class correctly uses the Data interface and Info class?

A.
```
public class Employee extends Info implements Data { public void
load() { /*do something*/ }
}
```
B.
```
public class Employee implements Info extends Data { public void
load() { /*do something*/ }
}
```
C.
```
public class Employee extends Info implements Data public void
load(){ /*do something*/ }
public void Info.load(){ /*do something*/ }
}
```
D.
```
public class Employee implements Info extends Data { public void
Data.load(){ /*do something*/ }
public void load(){ /*do something*/ }
}
```
E.
```
public class Employee implements Info extends Data { public void
load(){ /*do something*/ }
public void Info.load(){ /*do something*/ }
}
```
F.
```
public class Employee extends Info implements Data{ public void
Data.load() { /*do something*/ }
public void Info.load() { /*do something*/ }
}
```

Answer: A
Section: All

Explanation/Reference:

QUESTION 12
Given:

```
1. class Alligator {
2.     public static void main(String[] args) {
3.         int[] x[] = { { 1, 2 }, { 3, 4, 5 }, { 6, 7, 8, 9 } };
4.         int[][] y = x;
5.         System.out.println(y[2][1]);
6.     }
7. }
```

What is the result?

A. 2
B. 3
C. 4
D. 6
E. 7
F. Compilation fails.

Answer: E
Section: All

Explanation/Reference:
7

QUESTION 13
Given:

```
abstract class C1 {
public C1() { System.out.print(1); }
}
class C2 extends C1 {
public C2() { System.out.print(2); }
}
class C3 extends C2 {
public C3() { System.out.println(3); }
}
public class Ctest {
public static void main(String[] a) { new C3(); }
}
```

What is the result?

A. 3
B. 23
C. 32
D. 123
E. 321
F. Compilation fails.
G. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:
123

QUESTION 14
Given:

```
1. class One {
2.     public One foo() {
3.         return this;
4.     }
5. }
6.
7. class Two extends One {
8.     public One foo() {
9.         return this;
10.    }
11.}
12.
13.class Three extends Two {
14.    // insert method here
15.}
```

Which two methods, inserted individually, correctly complete the Three class? (Choose two.)

A. `public void foo() {}`
B. `public int foo() { return 3; }`
C. `public Two foo() { return this; }`
D. `public One foo() { return this; }`
E. `public Object foo() { return this; }`

Answer: CD
Section: All

Explanation/Reference:
A.
```
Main.java:14: foo() in Three cannot override foo() in Two; attempting to use incompatible return
type
found   : void
required: One
   public void foo() {}
               ^
1 error
```

B.
```
Main.java:14: foo() in Three cannot override foo() in Two; attempting to use incompatible return
type
found   : int
required: One
   public int foo() { return 3; }
              ^
1 error
```

C.
```
compiled successfully
```

D.
```
compiled successfully
```

E.

```
Main.java:14: foo() in Three cannot override foo() in Two; attempting to use incompatible return
type
found   : java.lang.Object
required: One
    public Object foo() { return this; }
                  ^
1 error
```

QUESTION 15
Which two classes correctly implement both the `java.lang.Runnable` and the `java.lang.Cloneable` interfaces? (Choose two.)

A.
```
public class Session implements Runnable, Cloneable {
    public void run();
    public Object clone();
}
```
B.
```
public class Session
extends Runnable, Cloneable {
    public void run() { /* do something */ }
    public Object clone() { /* make a copy */ }
}
```
C.
```
public class Session
implements Runnable, Cloneable {
    public void run() { /* do something */ }
    public Object clone() { /* make a copy */ }
}
```
D.
```
public abstract class Session
implements Runnable, Cloneable {
    public void run() { /* do something */ }
    public Object clone() { /*make a copy */ }
}
```
E.
```
public class Session
implements Runnable, implements Cloneable {
    public void run() { /* do something */ }
    public Object clone() { /* make a copy */ }
}
```

Answer: CD
Section: All

Explanation/Reference:

QUESTION 16
Given:

```
1. class TestA {
2.     public void start() { System.out.println("TestA"); }
3. }
4. public class TestB extends TestA {
5.     public void start() { System.out.println("TestB"); }
6.     public static void main(String[] args) {
7.         ((TestA)new TestB()).start();
8.     }
9. }
```

What is the result?

A. `TestA`
B. `TestB`
C. Compilation fails.
D. An exception is thrown at runtime.

Answer: B
Section: All

Explanation/Reference:
`TestB`

QUESTION 17
Given:

```
11. public static void main(String[] args) {
12.    Object obj = new int[] { 1, 2, 3 };
13.    int[] someArray = (int[])obj;
14.    for (int i : someArray) System.out.print(i + " ");
15. }
```

What is the result?

A. `1 2 3`
B. Compilation fails because of an error in line 12.
C. Compilation fails because of an error in line 13.
D. Compilation fails because of an error in line 14.
E. A ClassCastException is thrown at runtime.

Answer: A
Section: All

Explanation/Reference:
`1 2 3`

QUESTION 18
Click the Exhibit button.

```
1. public class Threads1 {
2.     int x = 0;
3.     public class Runner implements Runnable {
4.         public void run(){
5.             int current = 0;
6.             for(int i = 0; i<4; i++){
7.                 current = x;
8.                 System.out.println(current + ", ");
9.                 x = current + 2;
10.            }
11.        }
12.    }
13.
14.    public static void main(String[] args) {
15.        new Threads1().go();
16.    }
17.
```

```
18.    public void go(){
19.       Runnable r1 = new Runner();
20.       new Thread(r1).start();
21.       new Thread(r1).start();
22.    }
23. }
```

Which two are possible results? (Choose two.)

A. `0, 2, 4, 4, 6, 8, 10, 6,`
B. `0, 2, 4, 6, 8, 10, 2, 4,`
C. `0, 2, 4, 6, 8, 10, 12, 14,`
D. `0, 0, 2, 2, 4, 4, 6, 6, 8, 8, 10, 10, 12, 12, 14, 14,`
E. `0, 2, 4, 6, 8, 10, 12, 14, 0, 2, 4, 6, 8, 10, 12, 14,`

Answer: AC
Section: All

Explanation/Reference:
`A. 0, 2, 4, 4, 6, 8, 10, 6,`
`  1  1  1  2  2  2   2  1` <- possible threads that produced this output - possible solution

`B. 0, 2, 4, 6, 8, 10, 2, 4,`
`  1  2  2  2  2   ?  1` <- to print second '2', T1 interrupted between L10/L11; 4 passes of T2 used up

`C. 0, 2, 4, 6, 8, 10, 12, 14,`
`  1  1  1  1  2   2   2   2` <- possible solution - simplest solution (T2 waits until T1 is completely done) - doesn't matter that it isn't likely, just that is possible

`D. 0, 0, 2, 2, 4, 4, 6, 6, 8, 8, 10, 10, 12, 12, 14, 14,`
`  1  2  1  2  1  2  1  2  1  2   ?` <- threads used up

`E. 0, 2, 4, 6, 8, 10, 12, 14, 0, 2, 4, 6, 8, 10, 12, 14,`
`  1  1  1  1  2   2   2   2  ?` <- threads used up

QUESTION 19
Given:

`foo` and `bar` are public references available to many other threads. `foo` refers to a `Thread` and `bar` is an `Object`. The thread `foo` is currently executing `bar.wait()`. From another thread, what provides the most reliable way to ensure that `foo` will stop executing `wait()`?

A. `foo.notify();`
B. `bar.notify();`
C. `foo.notifyAll();`
D. `Thread.notify();`
E. `bar.notifyAll();`
F. `Object.notify();`

Answer: E
Section: All

Explanation/Reference:

QUESTION 20

Given:

```
public class PingPong implements Runnable {
    synchronized void hit(long n) {
        for (int i = 1; i < 3; i++)
            System.out.print(n + "-" + i + " ");
    }

    public static void main(String[] args) {
        new Thread(new PingPong()).start();
        new Thread(new PingPong()).start();
    }

    public void run() {
        hit(Thread.currentThread().getId());
    }
}
```

Which two statements are true? (Choose two.)

A. The output could be `8-1 7-2 8-2 7-1`
B. The output could be `7-1 7-2 8-1 6-1`
C. The output could be `8-1 7-1 7-2 8-2`
D. The output could be `8-1 8-2 7-1 7-2`

Answer: CD
Section: All

Explanation/Reference:

QUESTION 21
Click the Exhibit button.

```
class Computation extends Thread {

    private int num;
    private boolean isComplete;
    private int result;

    public Computation(int num){ this.num = num; }

    public synchronized void run() {
        result = num * 2;
        isComplete = true;
        notify();
    }

    public synchronized int getResult() {
        while ( ! isComplete ){
            try {
                wait();
            } catch (InterruptedException e) {
            }
        }
        return result;
    }

    public static void main(String[] args) {
```

```
        Computation[] computations = new Computation[4];
        for (int i = 0; i < computations.length; i++) {
            computations[i] = new Computation(i);
            computations[i].start();
        }
        for (Computation c : computations) {
            System.out.println(c.getResult() + " ");
        }
    }
}
```

What is the result?

A. The code will deadlock.
B. The code may run with no output.
C. An exception is thrown at runtime.
D. The code may run with output "`0 6`".
E. The code may run with output "`2 0 6 4`".
F. The code may run with output "`0 2 4 6`".

Answer: F
Section: All

Explanation/Reference:
`0 2 4 6`

QUESTION 22
Which two code fragments will execute the method `doStuff()` in a separate thread? (Choose two.)

A.
```
new Thread() {
    public void run() { doStuff(); }
};
```
B.
```
new Thread() {
    public void start() { doStuff(); }
};
```
C.
```
new Thread() {
    public void start() { doStuff(); }
}.run();
```
D.
```
new Thread() {
    public void run() { doStuff(); }
}.start();
```
E.
```
new Thread(new Runnable() {
    public void run() { doStuff(); }
}).run();
```
F.
```
new Thread(new Runnable() {
    public void run() { doStuff(); }
}).start();
```

Answer: DF
Section: All

Explanation/Reference:

QUESTION 23
Given:

```
public class Person {
    private String name;
    public Person(String name) {
        this.name = name;
    }
    public boolean equals(Object o) {
        if ( ! ( o instanceof Person) ) return false;
        Person p = (Person) o;
        return p.name.equals(this.name);
    }
}
```

Which statement is true?

A. Compilation fails because the `hashCode` method is not overridden.
B. A `HashSet` could contain multiple `Person` objects with the same name.
C. All `Person` objects will have the same hash code because the `hashCode` method is not overridden.
D. If a `HashSet` contains more than one `Person` object with name="Fred", then removing another `Person`, also with name="Fred", will remove them all.

Answer: B
Section: All

Explanation/Reference:

QUESTION 24
Given:

```
import java.util.*;
public class SortOf {
    public static void main(String[] args) {
        ArrayList<Integer> a = new ArrayList<Integer>();
        a.add(1); a.add(5); a.add(3);
        Collections.sort(a);
        a.add(2);
        Collections.reverse(a);
        System.out.println(a);
    }
}
```

What is the result?

A. `[1, 2, 3, 5]`
B. `[2, 1, 3, 5]`
C. `[2, 5, 3, 1]`
D. `[5, 3, 2, 1]`
E. `[1, 3, 5, 2]`
F. Compilation fails.
G. An exception is thrown at runtime.

Answer: C

Section: All

Explanation/Reference:
[2, 5, 3, 1]

QUESTION 25
Given:

```
public class Person {
    private name;
    public Person(String name) {
        this.name = name;
    }
    public int hashCode() {
        return 420;
    }
}
```

Which statement is true?

A. The time to find the value from `HashMap` with a `Person` key depends on the size of the map.
B. Deleting a `Person` key from a `HashMap` will delete all map entries for all keys of type `Person`.
C. Inserting a second `Person` object into a `HashSet` will cause the first `Person` object to be removed as a duplicate.
D. The time to determine whether a `Person` object is contained in a `HashSet` is constant and does NOT depend on the size of the map.

Answer: A
Section: All

Explanation/Reference:

QUESTION 26
Given:

```
public class Drink implements Comparable {
    public String name;
    public int compareTo(Object o) {
        return 0;
    }
}
```

and:

```
Drink one = new Drink();
Drink two = new Drink();
one.name= "Coffee";
two.name= "Tea";
TreeSet set = new TreeSet();
set.add(one);
set.add(two);
```

A programmer iterates over the `TreeSet` and prints the name of each `Drink` object.
What is the result?

A. `Tea`
B. `Coffee`
C. `Coffee`
`Tea`
D. Compilation fails.
E. The code runs with no output.
F. An exception is thrown at runtime.

Answer: B
Section: All

Explanation/Reference:

QUESTION 27
A programmer must create a generic class `MinMax` and the type parameter of `MinMax` must implement `Comparable`. Which implementation of `MinMax` will compile?

A.
```
class MinMax<E extends Comparable<E>> {
    E min = null;
    E max = null;
    public MinMax() {}
    public void put(E value) { /* store min or max */ }
```
B.
```
class MinMax<E implements Comparable<E>> {
    E min = null;
    E max = null;
    public MinMax() {}
    public void put(E value) { /* store min or max */ }
```
C.
```
class MinMax<E extends Comparable<E>> {
    <E> E min = null;
    <E> E max = null;
    public MinMax() {}
    public <E> void put(E value) { /* store min or max */ }
```
D.
```
class MinMax<E implements Comparable<E>> {
    <E> E min = null;
    <E> E max = null;
    public MinMax() {}
    public <E> void put(E value) { /* store min or max */ }
```

Answer: A
Section: All

Explanation/Reference:

QUESTION 28
Given:

```
1. import java.util.*;
2. public class Example {
3.     public static void main(String[] args) {
4.         // insert code here
5.         set.add(new Integer(2));
6.         set.add(new Integer(1));
7.         System.out.println(set);
8.     }
9. }
```

Which code, inserted at line 4, guarantees that this program will output `[1, 2]`?

A. `Set set = new TreeSet();`
B. `Set set = new HashSet();`
C. `Set set = new SortedSet();`
D. `List set = new SortedList();`
E. `Set set = new LinkedHashSet();`

Answer: A
Section: All

Explanation/Reference:

QUESTION 29
Given:

```
05. class A {
06.    void foo() throws Exception { throw new Exception(); }
07. }
08. class SubB2 extends A {
09.     void foo() { System.out.println("B "); }
10. }
11. class Tester {
12.    public static void main(String[] args) {
13.       A a = new SubB2();
14.       a.foo();
15.    }
16. }
```

What is the result?

A. `B`
B. `B`, followed by an Exception.
C. Compilation fails due to an error on line 9.
D. Compilation fails due to an error on line 14.
E. An Exception is thrown with no other output.

Answer: D
Section: All

Explanation/Reference:

```
Main.java:14: unreported exception java.lang.Exception; must be caught or declared to be thrown
      a.foo();
          ^
1 error
```

QUESTION 30
Given:

```
1. public class Breaker {
2.     static String o = "";
3.
4.     public static void main(String[] args) {
5.         z: o = o + 2;
6.         for (int x = 3; x < 8; x++) {
```

```
7.            if (x == 4)
8.               break;
9.            if (x == 6)
10.              break z;
11.           o = o + x;
12.       }
13.       System.out.println(o);
14.   }
15. }
```

What is the result?

A. 23
B. 234
C. 235
D. 2345
E. 2357
F. 23457
G. Compilation fails.

Answer: G
Section: All

Explanation/Reference:

```
Main.java:10: undefined label: z
            break z;
            ^
1 error
```

QUESTION 31
Given:

```
11.   public void go(int x) {
12.      assert (x > 0);
13.      switch(x) {
14.      case 2: ;
15.      default: assert false;
16.      }
17.   }
18.   private void go2(int x) { assert (x < 0); }
```

Which statement is true?

A. All of the assert statements are used appropriately.
B. Only the assert statement on line 12 is used appropriately.
C. Only the assert statement on line 15 is used appropriately.
D. Only the assert statement on line 18 is used appropriately.
E. Only the assert statements on lines 12 and 15 are used appropriately.
F. Only the assert statements on lines 12 and 18 are used appropriately.
G. Only the assert statements on lines 15 and 18 are used appropriately.

Answer: G
Section: All

Explanation/Reference:

QUESTION 32
Given:

```
1.     public static void main(String[] args) {
2.         try {
3.             args = null;
4.             args[0] = "test";
5.             System.out.println(args[0]);
6.         } catch (Exception ex) {
7.             System.out.println("Exception");
8.         } catch (NullPointerException npe) {
9.             System.out.println("NullPointerException");
10.        }
11.    }
```

What is the result?

A. `test`
B. Exception
C. Compilation fails.
D. `NullPointerException`

Answer: C
Section: All

Explanation/Reference:

```
Main.java:8: exception java.lang.NullPointerException has already been caught
        } catch (NullPointerException npe) {
          ^
1 error
```

QUESTION 33
Given:

```
1.     public static void main(String[] args) {
2.         for (int i = 0; i <= 10; i++) {
3.             if (i > 6) break;
4.         }
5.         System.out.println(i);
6.     }
```

What is the result?

A. `6`
B. `7`
C. `10`
D. `11`
E. Compilation fails.
F. An exception is thrown at runtime.

Answer: E
Section: All

Explanation/Reference:

```
Main.java:5: cannot find symbol
symbol  : variable i
```

```
location: class Breaker
       System.out.println(i);
                          ^
1 error
```

QUESTION 34
Given:

```
11. public void testIfA() {
12.     if (testIfB("True")) {
13.         System.out.println("True");
14.     } else {
15.         System.out.println("Not true");
16.     }
17. }
18. public Boolean testIfB(String str) {
19.     return Boolean.valueOf(str);
20. }
```

What is the result when method testIfA is invoked?

A. `True`
B. `Not true`
C. An exception is thrown at runtime.
D. Compilation fails because of an error at line 12.
E. Compilation fails because of an error at line 19.

Answer: A
Section: All

Explanation/Reference:
`True`

QUESTION 35
Which can appropriately be thrown by a programmer using Java SE technology to create a desktop application?

A. `ClassCastException`
B. `NullPointerException`
C. `NoClassDefFoundError`
D. `NumberFormatException`
E. `ArrayIndexOutOfBoundsException`

Answer: D
Section: All

Explanation/Reference:

QUESTION 36
Which two code fragments are most likely to cause a StackOverflowError? (Choose two.)

A.
```
int []x = {1,2,3,4,5};
for(int y = 0; y < 6; y++)
    System.out.println(x[y]);
```
B.
```
static int[] x = {7,6,5,4};
```

```
   static { x[1] = 8;
   x[4] = 3; }
```
C.
```
for(int y = 10; y < 10; y++)
    doStuff(y);
```
D.
```
void doOne(int x) { doTwo(x); }
void doTwo(int y) { doThree(y); }
void doThree(int z) { doTwo(z); }
```
E.
```
for(int x = 0; x < 1000000000; x++)
    doStuff(x);
```
F.
```
void counter(int i) { counter(++i); }
```

Answer: DF
Section: All

Explanation/Reference:

QUESTION 37
Given:

```
04. public class Tahiti {
05.     Tahiti t;
06.
07.     public static void main(String[] args) {
08.         Tahiti t = new Tahiti();
09.         Tahiti t2 = t.go(t);
10.         t2 = null;
11.         // more code here
12.     }
13.
14.     Tahiti go(Tahiti t) {
15.         Tahiti t1 = new Tahiti();
16.         Tahiti t2 = new Tahiti();
17.         t1.t = t2;
18.         t2.t = t1;
19.         t.t = t2;
20.         return t1;
21.     }
22. }
```

When line 11 is reached, how many objects are eligible for garbage collection?

A. 0
B. 1
C. 2
D. 3
E. Compilation fails.

Answer: A
Section: All

Explanation/Reference:

QUESTION 38
Given:

```
interface Animal {
    void makeNoise();
}

class Horse implements Animal {
    Long weight = 1200L;

    public void makeNoise() {
        System.out.println("whinny");
    }
}

01. public class Icelandic extends Horse {
02.     public void makeNoise() {
03.         System.out.println("vinny");
04.     }
05.
06.     public static void main(String[] args) {
07.         Icelandic i1 = new Icelandic();
08.         Icelandic i2 = new Icelandic();
09.         Icelandic i3 = new Icelandic();
10.         i3 = i1;
11.         i1 = i2;
12.         i2 = null;
13.         i3 = i1;
14.     }
15. }
```

When line 14 is reached, how many objects are eligible for the garbage collector?

A. 0
B. 1
C. 2
D. 3
E. 4
F. 6

Answer: E
Section: All

Explanation/Reference:

QUESTION 39
Given:

```
11. public class Commander {
12.     public static void main(String[] args) {
13.         String myProp = /* insert code here */
14.         System.out.println(myProp);
15.     }
16. }
```

and the command line: **`java -Dprop.custom=gobstopper Commander`**

Which two, placed on line 13, will produce the output gobstopper? (Choose two.)

A. `System.load("prop.custom");`
B. `System.getenv("prop.custom");`

C. `System.property("prop.custom");`
D. `System.getProperty("prop.custom");`
E. `System.getProperties().getProperty("prop.custom");`

Answer: DE
Section: All

Explanation/Reference:

QUESTION 40
Given:

```
1. public class ItemTest {
2.     private final int id;
3.
4.     public ItemTest(int id) {
5.         this.id = id;
6.     }
7.
8.     public void updateId(int newId) {
9.         id = newId;
10.    }
11.
12.    public static void main(String[] args) {
13.        ItemTest fa = new ItemTest(42);
14.        fa.updateId(69);
15.        System.out.println(fa.id);
16.    }
17.}
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. The attribute `id` in the `ItemTest` object remains unchanged.
D. The attribute `id` in the `ItemTest` object is modified to the new value.
E. A new `ItemTest` object is created with the preferred value in the `id` attribute.

Answer: A
Section: All

Explanation/Reference:

```
Main.java:9: cannot assign a value to final variable id
       id = newId;
       ^
1 error
```

QUESTION 41
A developer is creating a class `Book`, that needs to access class `Paper`.
The `Paper` class is deployed in a JAR named `myLib.jar`.
Which three, taken independently, will allow the developer to use the `Paper` class while compiling the `Book` class? (Choose three.)

A. The JAR file is located at $JAVA_HOME/jre/classes/myLib.jar.

B. The JAR file is located at $JAVA_HOME/jre/lib/ext/myLib.jar..
C. The JAR file is located at /foo/myLib.jar and a classpath environment variable is set that includes /foo/myLib.jar/Paper.class.
D. The JAR file is located at /foo/myLib.jar and a classpath environment variable is set that includes /foo/myLib.jar.
E. The JAR file is located at /foo/myLib.jar and the Book class is compiled using javac -cp /foo/myLib.jar/Paper Book.java.
F. The JAR file is located at /foo/myLib.jar and the Book class is compiled using javac -d /foo/myLib.jar Book.java
G. The JAR file is located at /foo/myLib.jar and the Book class is compiled using javac -classpath /foo/myLib.jar Book.java

Answer: BDG
Section: All

Explanation/Reference:

QUESTION 42
Click the Exhibit button.

```
class Foo {
    private int x;
    public Foo( int x ){ this.x = x;}
    public void setX( int x ) { this.x = x; }
    public int getX(){ return x;}
}

public class Gamma {

    static Foo fooBar(Foo foo) {
        foo = new Foo(100);
        return foo;
    }

    public static void main(String[] args) {
        Foo foo = new Foo( 300 );
        System.out.println( foo.getX() + "-");

        Foo fooFoo = fooBar(foo);
        System.out.println(foo.getX() + "-");
        System.out.println(fooFoo.getX() + "-");

        foo = fooBar( fooFoo);
        System.out.println( foo.getX() + "-");
        System.out.println(fooFoo.getX());
    }
}
```

What is the output of the program shown in the exhibit?

A. 300-100-100-100-100
B. 300-300-100-100-100
C. 300-300-300-100-100
D. 300-300-300-300-100

Answer: B

Section: All

Explanation/Reference:
300-300-100-100-100

QUESTION 43
Given classes defined in two different files:

```
1. package packageA;
2. public class Message {
3.     String getText() {
4.         return "text";
5.     }
6. }
```

And:

```
1. package packageB;
2.
3. public class XMLMessage extends packageA.Message {
4.     String getText() {
5.         return "<msg>text</msg>";
6.     }
7.
8.     public static void main(String[] args) {
9.         System.out.println(new XMLMessage().getText());
10.    }
11. }
```

What is the result of executing `XMLMessage.main`?

A. `text`
B. Compilation fails.
C. `<msg>text</msg>`
D. An exception is thrown at runtime.

Answer: C
Section: All

Explanation/Reference:
<msg>text</msg>

QUESTION 44
Given:

```
interface Fish {
}

class Perch implements Fish {
}

class Walleye extends Perch {
}

class Bluegill {
}
```

```
public class Fisherman {
    public static void main(String[] args) {
        Fish f = new Walleye();
        Walleye w = new Walleye();
        Bluegill b = new Bluegill();
        if (f instanceof Perch)
            System.out.print("f-p ");
        if (w instanceof Fish)
            System.out.print("w-f ");
        if (b instanceof Fish)
            System.out.print("b-f ");
    }
}
```

What is the result?

A. `w-f`
B. `f-p w-f`
C. `w-f b-f`
D. `f-p w-f b-f`
E. Compilation fails.
F. An exception is thrown at runtime.

Answer: B
Section: All

Explanation/Reference:
`f-p w-f`

QUESTION 45
Given:

```
1. package com.company.application;
2.
3. public class MainClass {
4.     public static void main(String[] args) {
5.     }
6. }
```

And `MainClass` exists in the `/apps/com/company/application` directory.
Assume the CLASSPATH environment variable is set to "." (current directory).
Which two java commands entered at the command line will run `MainClass`? (Choose two.)

A. `java MainClass` if run from the `/apps` directory
B. `java com.company.application.MainClass` if run from the `/apps` directory
C. `java -classpath /apps com.company.application.MainClass` if run from any directory
D. `java -classpath . MainClass` if run from the `/apps/com/company/`application directory
E. `java -classpath /apps/com/company/application:. MainClass` if run from the `/apps` directory
F. `java com.company.application.MainClass` if run from the `/apps/com/company/application` directory

Answer: BC
Section: All

Explanation/Reference:

QUESTION 46
Given

```
class Foo {
    static void alpha() {
        /* more code here */
    }

    void beta() {
        /* more code here */
    }
}
```

Which two statements are true? (Choose two.)

A. `Foo.beta()` is a valid invocation of `beta()`.
B. `Foo.alpha()` is a valid invocation of `alpha()`.
C. Method `beta()` can directly call method `alpha()`.
D. Method `alpha()` can directly call method `beta()`.

Answer: BC
Section: All

Explanation/Reference:

QUESTION 47
Given:

```
1. public class TestSeven extends Thread {
2.     private static int x;
3.     public synchronized void doThings() {
4.         int current = x;
5.         current++;
6.         x = current;
7.     }
8.     public void run() {
9.         doThings();
10.    }
11.}
```

Which statement is true?

A. Compilation fails.
B. An exception is thrown at runtime.
C. Synchronizing the `run()` method would make the class thread-safe.
D. The data in variable `x` are protected from concurrent access problems.
E. Declaring the `doThings()` method as `static` would make the class thread-safe.
F. Wrapping the statements within `doThings()` in a `synchronized(new Object()) { }` block would make the class thread-safe.

Answer: E
Section: All

Explanation/Reference:

QUESTION 48
Given that the current directory is empty, and that the user has read and write privileges to the current directory, and the following:

```
1. import java.io.*;
2. public class Maker {
3.     public static void main(String[] args) {
4.         File dir = new File("dir");
5.         File f = new File(dir, "f");
6.     }
7. }
```

Which statement is true?

A. Compilation fails.
B. Nothing is added to the file system.
C. Only a new file is created on the file system.
D. Only a new directory is created on the file system.
E. Both a new file and a new directory are created on the file system.

Answer: B
Section: All

Explanation/Reference:

QUESTION 49
Given:

```
NumberFormat nf = NumberFormat.getInstance();
nf.setMaximumFractionDigits(4);
nf.setMinimumFractionDigits(2);
String a = nf.format(3.1415926);
String b = nf.format(2);
```

Which two statements are true about the result if the default locale is Locale.US? (Choose two.)

A. The value of `b` is `2`.
B. The value of `a` is `3.14`.
C. The value of `b` is `2.00`.
D. The value of `a` is `3.141`.
E. The value of `a` is `3.1415`.
F. The value of `a` is `3.1416`.
G. The value of `b` is `2.0000`.

Answer: CF
Section: All

Explanation/Reference:

```
a = 3.1416
b = 2.00
```

QUESTION 50
Which three statements concerning the use of the **`java.io.Serializable`** interface are true? (Choose three.)

A. Objects from classes that use aggregation cannot be serialized.
B. An object serialized on one JVM can be successfully deserialized on a different JVM.
C. The values in fields with the volatile modifier will NOT survive serialization and deserialization.
D. The values in fields with the transient modifier will NOT survive serialization and deserialization.
E. It is legal to serialize an object of a type that has a supertype that does NOT implement java.io.Serializable.

Answer: BDE
Section: All

Explanation/Reference:

QUESTION 51
Given:

```
12. String csv = "Sue,5,true,3";
13. Scanner scanner = new Scanner( csv );
14. scanner.useDelimiter(",");
15. int age = scanner.nextInt();
```

What is the result?

A. Compilation fails.
B. After line 15, the value of age is 5.
C. After line 15, the value of age is 3.
D. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:

```
Exception in thread "main" java.util.InputMismatchException
    at java.util.Scanner.throwFor(Scanner.java:840)
    at java.util.Scanner.next(Scanner.java:1461)
    at java.util.Scanner.nextInt(Scanner.java:2091)
    at java.util.Scanner.nextInt(Scanner.java:2050)
    at main(Main.java:15)
```

QUESTION 52
Given that c is a reference to a valid java.io.Console object, which two code fragments read a line of text from the console? (Choose two.)

A. `String s = c.readLine();`
B. `char[] c = c.readLine();`
C. `String s = c.readConsole();`
D. `char[] c = c.readConsole();`
E. `String s = c.readLine("%s", "name ");`
F. `char[] c = c.readLine("%s", "name ");`

Answer: AE
Section: All

Explanation/Reference:

QUESTION 53
Given:

```
11. String test = "a1b2c3";
12. String[] tokens = test.split("\\d");
13. for(String s: tokens) System.out.print(s + " ");
```

What is the result?

A. `a b c`
B. `1 2 3`
C. `a1b2c3`
D. `a1 b2 c3`
E. Compilation fails.
F. The code runs with no output.
G. An exception is thrown at runtime.

Answer: A
Section: All

Explanation/Reference:
`a b c`

QUESTION 54
Given:

```
33. Date d = new Date(0);
34. String ds = "December 15, 2004";
35. // insert code here
36. try {
37.     d = df.parse(ds);
38. }
39. catch(ParseException e) {
40.     System.out.println("Unable to parse " + ds);
41. }
42. // insert code here too
```

What creates the appropriate DateFormat object and adds a day to the Date object?

A.
```
35. DateFormat df = DateFormat.getDateFormat();
42. d.setTime( (60 * 60 * 24) + d.getTime());
```
B.
```
35. DateFormat df = DateFormat.getDateInstance();
42. d.setTime( (1000 * 60 * 60 * 24) + d.getTime());
```
C.
```
35. DateFormat df = DateFormat.getDateFormat();
42. d.setLocalTime( (1000*60*60*24) + d.getLocalTime());
```
D.
```
35. DateFormat df = DateFormat.getDateInstance();
42. d.setLocalTime( (60 * 60 * 24) + d.getLocalTime());
```

Answer: B
Section: All

Explanation/Reference:

QUESTION 55
Given:

```
import java.util.*;
public class Quest {
    public static void main(String[] args) {
        String[] colors = {"blue", "red", "green", "yellow", "orange"};
        Arrays.sort(colors);
        int s2 = Arrays.binarySearch(colors, "orange");
        int s3 = Arrays.binarySearch(colors, "violet");
        System.out.println(s2 + " " + s3);
    }
}
```

What is the result?

A. `2 -1`
B. `2 -4`
C. `2 -5`
D. `3 -1`
E. `3 -4`
F. `3 -5`
G. Compilation fails.
H. An exception is thrown at runtime.

Answer: C
Section: All

Explanation/Reference:
`2 -5`

QUESTION 56
Given:

```
static void test() {
    try {
        String x = null;
        System.out.print(x.toString() + " ");
    }
    finally { System.out.print("finally "); }
}
public static void main(String[] args) {
    try { test(); }
    catch (Exception ex) { System.out.print("exception "); }
}
```

What is the result?

A. `null`
B. `finally`
C. `null finally`
D. Compilation fails.
E. `finally exception`

Answer: E
Section: All

Explanation/Reference:
finally exception

QUESTION 57
Given:

```
public static Collection get() {
    Collection sorted = new LinkedList();
    sorted.add("B"); sorted.add("C"); sorted.add("A");
    return sorted;
}
public static void main(String[] args) {
    for (Object obj: get()) {
        System.out.print(obj + ", ");
    }
}
```

What is the result?

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

Answer: B
Section: All

Explanation/Reference:
B, C, A,

QUESTION 58
Given:

```
public static Iterator reverse(List list) {
    Collections.reverse(list);
    return list.iterator();
}
public static void main(String[] args) {
    List list = new ArrayList();
    list.add("1"); list.add("2"); list.add("3");
    for (Object obj: reverse(list))
        System.out.print(obj + ", ");
}
```

What is the result?

A. 3, 2, 1,
B. 1, 2, 3,
C. Compilation fails.
D. The code runs with no output.
E. An exception is thrown at runtime.

Answer: C

Section: All

Explanation/Reference:

QUESTION 59
Given:

```
public class Base {
    public static final String FOO = "foo";

    public static void main(String[] args) {
        Base b = new Base();
        Sub s = new Sub();
        System.out.print(Base.FOO);
        System.out.print(Sub.FOO);
        System.out.print(b.FOO);
        System.out.print(s.FOO);
        System.out.print(((Base) s).FOO);
    }
}

class Sub extends Base {
    public static final String FOO = "bar";
}
```

What is the result?

A. foofoofoofoofoo
B. foobarfoobarbar
C. foobarfoofoofoo
D. foobarfoobarfoo
E. barbarbarbarbar
F. foofoofoobarbar
G. foofoofoobarfoo

Answer: D
Section: All

Explanation/Reference:
foobarfoobarfoo

QUESTION 60
Given:

```
1. public interface A { public void m1(); }
2.
3. class B implements A { }
4. class C implements A { public void m1() { } }
5. class D implements A { public void m1(int x) { } }
6. abstract class E implements A { }
7. abstract class F implements A { public void m1() { } }
8. abstract class G implements A { public void m1(int x) { } }
```

What is the result?

A. Compilation succeeds.

B. Exactly one class does NOT compile.
C. Exactly two classes do NOT compile.
D. Exactly four classes do NOT compile.
E. Exactly three classes do NOT compile.

Answer: C
Section: All

Explanation/Reference:
```
Main.java:3: B is not abstract and does not override abstract method m1() in A
class B implements A { }
^
Main.java:5: D is not abstract and does not override abstract method m1() in A
class D implements A { public void m1(int x) { } }
^
2 errors
```

Exam D

QUESTION 1
Given:

```
1. public class KungFu {
2.     public static void main(String[] args) {
3.         Integer x = 400;
4.         Integer y = x;
5.         x++;
6.         StringBuilder sb1 = new StringBuilder("123");
7.         StringBuilder sb2 = sb1;
8.         sb1.append("5");
9.         System.out.println((x == y) + " " + (sb1 == sb2));
10.    }
11.}
```

What is the result?

A. `true true`
B. `false true`
C. `true false`
D. `false false`
E. Compilation fails.
F. An exception is thrown at runtime.

Answer: B
Section: All

Explanation/Reference:
```
false true
```

QUESTION 2
Given:

```
11. class Converter {
12.     public static void main(String[] args) {
13.         Integer i = args[0];
```

```
14.         int j = 12;
15.         System.out.println("It is " + (j == i) + " that j==i.");
16.     }
17. }
```

What is the result when the programmer attempts to compile the code and run it with the command `java Converter 12`?

A. It is true that j==i.
B. It is false that j==i.
C. An exception is thrown at runtime.
D. Compilation fails because of an error in line 13.

Answer: D
Section: All

Explanation/Reference:

```
Main.java:13: incompatible types
found   : java.lang.String
required: java.lang.Integer
       Integer i = args[0];
                   ^
1 error
```

QUESTION 3
Click the Exhibit button.

```
01. public class A {
02.     public String doit(int x, int y){
03.         return "a";
04.     }
05.
06.     public String doit(int... vals){
07.         return "b";
08.     }
09. }
```

Given:

```
25.  A a = new A();
26.  System.out.println(a.doit(4, 5));
```

What is the result?

A. Line 26 prints `a` to `System.out`.
B. Line 26 prints `b` to `System.out`.
C. An exception is thrown at line 26 at runtime.
D. Compilation of class `A` will fail due to an error in line 6.

Answer: A
Section: All

Explanation/Reference:

```
a
```

QUESTION 4
Given:

```
1. public class Plant {
2.     private String name;
3.
4.     public Plant(String name) {
5.         this.name = name;
6.     }
7.
8.     public String getName() {
9.         return name;
10.    }
11.}
```

```
1. public class Tree extends Plant {
2.     public void growFruit() {
3.     }
4.
5.     public void dropLeaves() {
6.     }
7. }
```

Which statement is true?

A. The code will compile without changes.
B. The code will compile if `public Tree() { Plant(); }` is added to the Tree class.
C. The code will compile if `public Plant() { Tree(); }` is added to the Plant class.
D. The code will compile if `public Plant() { this("fern"); }` is added to the Plant class.
E. The code will compile if `public Plant() { Plant("fern"); }` is added to the Plant class.

Answer: D
Section: All

Explanation/Reference:

QUESTION 5
Click the Exhibit button.

```
1. public class GoTest {
2.     public static void main(String[] args) {
3.         Sente a = new Sente(); a.go();
4.         Goban b = new Goban(); b.go();
5.         Stone c = new Stone(); c.go();
6.     }
7. }
8.
9. class Sente implements Go {
10.    public void go(){
11.        System.out.println("go in Sente");
12.    }
13.}
14.
15.class Goban extends Sente {
```

```
16.    public void go(){
17.        System.out.println("go in Goban");
18.    }
19.
20.}
21.class Stone extends Goban implements Go{
22.}
23.
24.interface Go { public void go(); }
```

What is the result?

A. go in Goban go in Sente go in Sente
B. go in Sente go in Sente go in Goban
C. go in Sente go in Goban go in Goban
D. go in Goban go in Goban go in Sente
E. Compilation fails because of an error in line 17.

Answer: C
Section: All

Explanation/Reference:

```
go in Sente
go in Goban
go in Goban
```

QUESTION 6
Given:

```
11. public interface A111 {
12.     String s = "yo";
13.     public void method1();
14. }
15.
16.
17. interface B {}
18.
19.
20. interface C extends A111, B {
21.     public void method1();
22.     public void method1(int x);
23. }
```

What is the result?

A. Compilation succeeds.
B. Compilation fails due to multiple errors.
C. Compilation fails due to an error only on line 20.
D. Compilation fails due to an error only on line 21.
E. Compilation fails due to an error only on line 22.
F. Compilation fails due to an error only on line 12.

Answer: A
Section: All

Explanation/Reference:

QUESTION 7
Click the Exhibit button.

```
10. interface Foo{
11.    int bar();
12. }
13.
14. public class Beta {
15.
16.     class A implements Foo {
17.        public int bar(){ return 1; }
18.     }
19.
20.     public int fubar(Foo foo){ return foo.bar(); }
21.
22.     public void testFoo(){
23.
24.        class A implements Foo{
25.           public int bar(){return 2;}
26.        }
27.
28.        System.out.println(fubar(new A()));
29.     }
30.
31.     public static void main(String[] args) {
32.        new Beta().testFoo();
33.     }
34. }
```

Which three statements are true? (Choose three.)

A. Compilation fails.
B. The code compiles and the output is `2`.
C. If lines 16, 17 and 18 were removed, compilation would fail.
D. If lines 24, 25 and 26 were removed, compilation would fail.
E. If lines 16, 17 and 18 were removed, the code would compile and the output would be `2`.
F. If lines 24, 25 and 26 were removed, the code would compile and the output would be `1`.

Answer: BEF
Section: All

Explanation/Reference:

QUESTION 8
Given:

```
1. public class TestOne {
2.     public static void main (String[] args) throws Exception {
3.        Thread.sleep(3000);
4.        System.out.println("sleep");
5.     }
6. }
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. The code executes normally and prints `sleep`.
D. The code executes normally, but nothing is printed.

Answer: C
Section: All

Explanation/Reference:
`sleep`

QUESTION 9
Given:

```
1. public class Threads3 implements Runnable {
2.     public void run() {
3.         System.out.print("running");
4.     }
5.     public static void main(String[] args) {
6.         Thread t = new Thread(new Threads3());
7.         t.run();
8.         t.run();
9.         t.start();
10.    }
11. }
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. The code executes and prints `running`.
D. The code executes and prints `runningrunning`.
E. The code executes and prints `runningrunningrunning`.

Answer: E
Section: All

Explanation/Reference:
`runningrunningrunning`

QUESTION 10
Given:

```
public class NamedCounter {
    private final String name;
    private int count;

    public NamedCounter(String name) {
        this.name = name;
    }

    public String getName() {
        return name;
    }
```

```
    public void increment() {
        count++;
    }

    public int getCount() {
        return count;
    }

    public void reset() {
        count = 0;
    }
}
```

Which three changes should be made to adapt this class to be used safely by multiple threads? (Choose three.)

A. declare `reset()` using the `synchronized` keyword
B. declare `getName()` using the `synchronized` keyword
C. declare `getCount()` using the `synchronized` keyword
D. declare the constructor using the `synchronized` keyword
E. declare `increment()` using the `synchronized` keyword

Answer: ACE
Section: All

Explanation/Reference:

QUESTION 11
Given that Triangle implements Runnable, and:

```
31.    void go() throws Exception {
32.        Thread t = new Thread(new Triangle());
33.        t.start();
34.        for(int x = 1; x < 100000; x++) {
35.            //insert code here
36.            if(x%100 == 0) System.out.print("g");
37.        } }
38.    public void run() {
39.        try {
40.            for(int x = 1; x < 100000; x++) {
41.                // insert the same code here
42.                if(x%100 == 0) System.out.print("t");
43.            }
44.        } catch (Exception e) {
45.
46.        }
47.    }
```

Which two statements, inserted independently at both lines 35 and 41, tend to allow both threads to temporarily pause and allow the other thread to execute? (Choose two.)

A. `Thread.wait();`
B. `Thread.join();`
C. `Thread.yield();`
D. `Thread.sleep(1);`
E. `Thread.notify();`

Answer: CD
Section: All

Explanation/Reference:

QUESTION 12
Given:

```
1. public class TestSeven extends Thread {
2.     private static int x;
3.     public synchronized void doThings() {
4.         int current = x;
5.         current++;
6.         x = current;
7.     }
8.     public void run() {
9.         doThings();
10.    }
11. }
```

Which statement is true?

A. Compilation fails.
B. An exception is thrown at runtime.
C. Synchronizing the `run()` method would make the class thread-safe.
D. The data in variable `x` are protected from concurrent access problems.
E. Declaring the `doThings()` method as `static` would make the class thread-safe.
F. Wrapping the statements within `doThings()` in a `synchronized(new Object()) { }` block would make the class thread-safe.

Answer: E
Section: All

Explanation/Reference:

QUESTION 13
Given:

```
public class Yikes {

    public static void go(Long n) {
        System.out.print("Long ");
    }

    public static void go(Short n) {
        System.out.print("Short ");
    }

    public static void go(int n) {
        System.out.print("int ");
    }

    public static void main(String[] args) {
        short y = 6;
```

```
        long z = 7;
        go(y);
        go(z);
    }
}
```

What is the result?

A. `int Long`
B. `Short Long`
C. Compilation fails.
D. An exception is thrown at runtime.

Answer: A
Section: All

Explanation/Reference:
`int Long`

QUESTION 14
Given:

```
12. Date date = new Date();
13. df.setLocale(Locale.ITALY);
14. String s = df.format(date);
```

The variable `df` is an object of type `DateFormat` that has been initialized in line 11.
What is the result if this code is run on `December 14, 2000`?

A. The value of s is 14-dic-2000.
B. The value of s is Dec 14, 2000.
C. An exception is thrown at runtime.
D. Compilation fails because of an error in line 13.

Answer: D
Section: All

Explanation/Reference:
The method `setLocale(Locale)` is undefined for the type `DateFormat`

QUESTION 15
Given that c is a reference to a valid java.io.Console object, and:

```
11. String pw = c.readPassword("%s", "pw: ");
12. System.out.println("got " + pw);
13. String name = c.readLine("%s", "name: ");
14. System.out.println(" got ", name);
```

If the user types fido when prompted for a password, and then responds bob when prompted for a name, what is the result?

A.
```
pw: got fido
name: bob got bob
```
B.
```
pw: fido got fido
name: bob got bob
```
C.
```
pw: got fido
name: bob got bob
```

D. `pw: fido got fido`
`name: bob got bob`
E. Compilation fails.
F. An exception is thrown at runtime.

Answer: E
Section: All

Explanation/Reference:
```
Main.java:14: cannot find symbol
symbol  : method println(java.lang.String,java.lang.String)
location: class java.io.PrintStream
       System.out.println(" got ", name);
                ^
1 error
```

QUESTION 16
Given:

```
11. String test = "This is a test";
12. String[] tokens = test.split("\s");
13. System.out.println(tokens.length);
```

What is the result?

A. 0
B. 1
C. 4
D. Compilation fails.
E. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:
```
Main.java:12: illegal escape character
String[] tokens = test.split("\s");
                             ^
1 error
```

QUESTION 17
Given:

```
import java.io.*;

class Animal {
    Animal() {
        System.out.print("a");
    }
}

class Dog extends Animal implements Serializable {
    Dog() {
        System.out.print("d");
    }
}

public class Beagle extends Dog {
```

```
}
```

If an instance of class Beagle is created, then Serialized, then deSerialized, what is the result?

A. `ad`
B. `ada`
C. `add`
D. `adad`
E. Compilation fails.
F. An exception is thrown at runtime.

Answer: B
Section: All

Explanation/Reference:

QUESTION 18
A team of programmers is involved in reviewing a proposed design for a new utility class. After some discussion, they realize that the current design allows other classes to access methods in the utility class that should be accessible only to methods within the utility class itself. What design issue has the team discovered?

A. Tight coupling
B. Low cohesion
C. High cohesion
D. Loose coupling
E. Weak encapsulation
F. Strong encapsulation

Answer: E
Section: All

Explanation/Reference:

QUESTION 19
Given a method that must ensure that its parameter is not null:

```
11. public void someMethod(Object value) {
12. // check for null value
...
20. System.out.println(value.getClass());
21. }
```

What, inserted at line 12, is the appropriate way to handle a null value?

A. `assert value == null;`
B. `assert value != null, "value is null";`
C. `if (value == null) { throw new AssertionException("value is null"); }`
D. `if (value == null) { throw new IllegalArgumentException("value is null"); }`

Answer: D
Section: All

Explanation/Reference:

QUESTION 20
Given:

```
1. public class Target {
2.     private int i = 0;
3.     public int addOne() {
4.         return ++i;
5.     }
6. }
```

And:

```
1. public class Client {
2.     public static void main(String[] args){
3.         System.out.println(new Target().addOne());
4.     }
5. }
```

Which change can you make to `Target` without affecting `Client`?

A. Line 4 of class `Target` can be changed to `return i++;`
B. Line 2 of class `Target` can be changed to `private int i = 1;`
C. Line 3 of class `Target` can be changed to `private int addOne(){`
D. Line 2 of class `Target` can be changed to `private Integer i = 0;`

Answer: D
Section: All

Explanation/Reference:

QUESTION 21
Given:

```
class Animal {
    public String noise() {
        return "peep";
    }
}

class Dog extends Animal {
    public String noise() {
        return "bark";
    }
}

class Cat extends Animal {
    public String noise() {
        return "meow";
    }
}
...

30. Animal animal = new Dog();
31. Cat cat = (Cat)animal;
```

```
32. System.out.println(cat.noise());
```

What is the result?

A. `peep`
B. `bark`
C. `meow`
D. Compilation fails.
E. An exception is thrown at runtime.

Answer: E
Section: All

Explanation/Reference:
```
Exception in thread "main" java.lang.ClassCastException: Dog cannot be cast to Cat
    at ClassTest.main(Main.java:31)
```

QUESTION 22
Given:

```
abstract class A {
    abstract void a1();

    void a2() {
    }
}

class B extends A {
    void a1() {
    }

    void a2() {
    }
}

class C extends B {
    void c1() {
    }
}
```

And:

```
A x = new B();
C y = new C();
A z = new C();
```

What are four valid examples of polymorphic method calls? (Choose four.)

A. `x.a2();`
B. `z.a2();`
C. `z.c1();`
D. `z.a1();`
E. `y.c1();`
F. `x.a1();`

Answer: ABDF
Section: All

Explanation/Reference:
A. `x.a2();` *// Method is overriden.*
B. `z.a2();` *// Method is inherited.*
C. `z.c1();` *// Won't compile. Method isn't defined in A.*
D. `z.a1();` *// Method is implemented.*
E. `y.c1();` *// Not polymorphic. Method is defined in declared class C.*
F. `x.a1();` *// Method is implemented.*

QUESTION 23
Given:

```
class Employee {
    String name;
    double baseSalary;

    Employee(String name, double baseSalary) {
        this.name = name;
        this.baseSalary = baseSalary;
    }
}

public class SalesPerson extends Employee {
    double commission;

    public SalesPerson(String name, double baseSalary, double commission) {
        // insert code here Line 13
    }
}
```

Which two code fragments, inserted independently at line 13, will compile? (Choose two.)

A. `super(name, baseSalary);`
B. `this.commission = commission;`
C. `super(); this.commission = commission;`
D. `this.commission = commission; super();`
E. `super(name, baseSalary); this.commission = commission;`
F. `this.commission = commission; super(name, baseSalary);`
G. `super(name, baseSalary, commission);`

Answer: AE
Section: All

Explanation/Reference:

QUESTION 24
A team of programmers is involved in reviewing a proposed design for a new utility class. After some discussion, they realize that the current design allows other classes to access methods in the utility class that should be accessible only to methods within the utility class itself.

What design issue has the team discovered?

A. Tight coupling
B. Low cohesion
C. High cohesion
D. Loose coupling
E. Weak encapsulation

F. Strong encapsulation

Answer: E
Section: All

Explanation/Reference:

QUESTION 25
Given that:
`Gadget` has-a `Sprocket` and `Gadget` has-a `Spring` and `Gadget` is-a `Widget` and `Widget` has-a `Sprocket`
Which two code fragments represent these relationships? (Choose two.)

A.
```
class Widget {
    Sprocket s;
}

class Gadget extends Widget {
    Spring s;
}
```
B.
```
class Widget {
}

class Gadget extends Widget {
    Spring s1;
    Sprocket s2;
}
```
C.
```
class Widget {
    Sprocket s1;
    Spring s2;
}

class Gadget extends Widget {
}
```
D.
```
class Gadget {
    Spring s;
}

class Widget extends Gadget {
    Sprocket s;
}
```
E.
```
class Gadget {
}

class Widget extends Gadget {
    Sprocket s1;
    Spring s2;
}
```
F.
```
class Gadget {
    Spring s1;
    Sprocket s2;
}

class Widget extends Gadget {
}
```

Answer: AC
Section: All

Explanation/Reference:

QUESTION 26
Given:

```
class Pizza {
    java.util.ArrayList toppings;

    public final void addTopping(String topping) {
        toppings.add(topping);
    }
    public void removeTopping(String topping) {
        toppings.remove(topping);
    }
}

public class PepperoniPizza extends Pizza {
    public void addTopping(String topping) {
        System.out.println("Cannot add Toppings");
    }

    public static void main(String[] args) {
        Pizza pizza = new PepperoniPizza();
        pizza.addTopping("Mushrooms");
        pizza.removeTopping("Peperoni");
    }
}
```

What is the result?

A. Compilation fails.
B. `Cannot add Toppings`
C. The code runs with no output.
D. A `NullPointerException` is thrown in Line 4.

Answer: A
Section: All

Explanation/Reference:

```
Main.java:13: addTopping(java.lang.String) in PepperoniPizza cannot override
addTopping(java.lang.String) in Pizza; overridden method is final
    public void addTopping(String topping) {
                ^
Note: Main.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
1 error
```

QUESTION 27
Which three statements are true? (Choose three.)

A. A `final` method in class `X` can be `abstract` if and only if `X` is `abstract`.
B. A `protected` method in class `X` can be overridden by any subclass of `X`.
C. A `private static` method can be called only within other `static` methods in class `X`.
D. A non-`static` `public final` method in class `X` can be overridden in any subclass of `X`.

E. A `public static` method in class `X` can be called by a subclass of `X` without explicitly referencing the class `X`.
F. A method with the same signature as a `private final` method in class `X` can be implemented in a subclass of `X`.
G. A `protected` method in class `X` can be overridden by a subclass of `X` only if the subclass is in the same `package` as `X`.

Answer: BEF
Section: All

Explanation/Reference:

QUESTION 28
Click the Exhibit button.

```
1. public class Car {
2.     private int wheelCount;
3.     private String vin;
4.     public Car(String vin){
5.         this.vin = vin;
6.         this.wheelCount = 4;
7.     }
8.     public String drive(){
9.         return "zoom-zoom";
10.    }
11.    public String getInfo() {
12.        return "VIN: " + vin + " wheels: " + wheelCount;
13.    }
14.}
```

And

```
1. public class MeGo extends Car {
2.     public MeGo(String vin) {
3.         this.wheelCount = 3;
4.     }
5. }
```

What two must the programmer do to correct the compilation errors? (Choose two.)

A. insert a call to `this()` in the `Car` constructor
B. insert a call to `this()` in the `MeGo` constructor
C. insert a call to `super()` in the `MeGo` constructor
D. insert a call to `super(vin)` in the `MeGo` constructor
E. change the `wheelCount` variable in `Car` to `protected`
F. change line 3 in the `MeGo` class to `super.wheelCount = 3;`

Answer: DE
Section: All

Explanation/Reference:

QUESTION 29

Click the Exhibit button.

```
1. import java.util.*;
2. public class TestSet{
3.     enum Example {ONE, TWO, THREE }
4.     public static void main(String[] args) {
5.         Collection coll = new ArrayList();
6.         coll.add(Example.THREE);
7.         coll.add(Example.THREE);
8.         coll.add(Example.THREE);
9.         coll.add(Example.TWO);
10.        coll.add(Example.TWO);
11.        coll.add(Example.ONE);
12.        Set set = new HashSet(coll);
13.    }
14. }
```

Which statement is true about the set variable on line 12?

A. The set variable contains all six elements from the coll collection, and the order is guaranteed to be preserved.
B. The set variable contains only three elements from the coll collection, and the order is guaranteed to be preserved.
C. The set variable contains all six elements from the coll collection, but the order is NOT guaranteed to be preserved.
D. The set variable contains only three elements from the coll collection, but the order is NOT guaranteed to be preserved.

Answer: D
Section: All

Explanation/Reference:

QUESTION 30
Given:

```
public class Person {
    private String name, comment;
    private int age;

    public Person(String n, int a, String c) {
        name = n;
        age = a;
        comment = c;
    }

    public boolean equals(Object o) {
        if (!(o instanceof Person))
            return false;
        Person p = (Person) o;
        return age == p.age && name.equals(p.name);
    }
}
```

What is the appropriate definition of the `hashCode` method in class `Person`?

A. `return super.hashCode();`
B. `return name.hashCode() + age * 7;`
C. `return name.hashCode() + comment.hashCode() / 2;`
D. `return name.hashCode() + comment.hashCode() / 2 - age * 3;`

Answer: B
Section: All

Explanation/Reference:

QUESTION 31
Given:

```
public class Key {
   private long id1;
   private long id2;

   // class Key methods
}
```

A programmer is developing a class `Key`, that will be used as a key in a standard `java.util.HashMap`.
Which two methods should be overridden to assure that `Key` works correctly as a key? (Choose two.)

A. `public int hashCode()`
B. `public boolean equals(Key k)`
C. `public int compareTo(Object o)`
D. `public boolean equals(Object o)`
E. `public boolean compareTo(Key k)`

Answer: AD
Section: All

Explanation/Reference:

QUESTION 32
A programmer has an algorithm that requires a `java.util.List` that provides an efficient implementation of `add(0, object)`, but does NOT need to support quick random access. What supports these requirements?

A. `java.util.Queue`
B. `java.util.ArrayList`
C. `java.util.LinearList`
D. `java.util.LinkedList`

Answer: D
Section: All

Explanation/Reference:

QUESTION 33
Given a class whose instances, when found in a collection of objects, are sorted by using the `compareTo()` method, which two statements are true? (Choose two.)

A. The class implements `java.lang.Comparable`.
B. The class implements `java.util.Comparator`.
C. The interface used to implement sorting allows this class to define only one sort sequence.
D. The interface used to implement sorting allows this class to define many different sort sequences.

Answer: AC
Section: All

Explanation/Reference:

QUESTION 34
Given:

```
1. import java.util.*;
2.
3. public class Explorer3 {
4.     public static void main(String[] args) {
5.         TreeSet<Integer> s = new TreeSet<Integer>();
6.         TreeSet<Integer> subs = new TreeSet<Integer>();
7.         for (int i = 606; i < 613; i++)
8.             if (i % 2 == 0)
9.                 s.add(i);
10.        subs = (TreeSet) s.subSet(608, true, 611, true);
11.        subs.add(629);
12.        System.out.println(s + " " + subs);
13.    }
14.}
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. `[608, 610, 612, 629] [608, 610]`
D. `[608, 610, 612, 629] [608, 610, 629]`
E. `[606, 608, 610, 612, 629] [608, 610]`
F. `[606, 608, 610, 612, 629] [608, 610, 629]`

Answer: B
Section: All

Explanation/Reference:

```
Exception in thread "main" java.lang.IllegalArgumentException: key out of range
    at java.util.TreeMap$NavigableSubMap.put(TreeMap.java:1386)
    at java.util.TreeSet.add(TreeSet.java:238)
    at Explorer3.main(Main.java:11)
```

QUESTION 35
Given:

```
1. import java.util.*;
2.
3. public class LetterASort {
```

```
4.     public static void main(String[] args) {
5.         ArrayList<String> strings = new ArrayList<String>();
6.         strings.add("aAaA");
7.         strings.add("AaA");
8.         strings.add("aAa");
9.         strings.add("AAaa");
10.        Collections.sort(strings);
11.        for (String s : strings) {
12.            System.out.print(s + " ");
13.        }
14.    }
15.}
```

What is the result?

A. Compilation fails.
B. `aAaA aAa AAaa AaA`
C. `AAaa AaA aAa aAaA`
D. `AaA AAaa aAaA aAa`
E. `aAa AaA aAaA AAaa`
F. An exception is thrown at runtime.

Answer: C
Section: All

Explanation/Reference:
`AAaa AaA aAa aAaA`

QUESTION 36
Given:

```
1. class A {
2.     void foo() throws Exception {
3.         throw new Exception();
4.     }
5. }
6.
7. class SubB2 extends A {
8.     void foo() {
9.         System.out.println("B ");
10.    }
11.}
12.class Tester {
13.    public static void main(String[] args) {
14.        A a = new SubB2();
15.        a.foo();
16.    }
17.}
```

What is the result?

A. `B`
B. `B`, followed by an Exception.
C. Compilation fails due to an error on line 9.
D. Compilation fails due to an error on line 15.
E. An Exception is thrown with no other output

Answer: D
Section: All

Explanation/Reference:
```
Main.java:15: unreported exception java.lang.Exception; must be caught or declared to be thrown
       a.foo();
           ^
1 error
```

QUESTION 37
Given a method that must ensure that its parameter is not null:

```
11. public void someMethod(Object value) {
12. // check for null value
...
20. System.out.println(value.getClass());
21. }
```

What, inserted at line 12, is the appropriate way to handle a null value?

A. `assert value == null;`
B. `assert value != null, "value is null";`
C. `if (value == null) { throw new AssertionException("value is null"); }`
D. `if (value == null) { throw new IllegalArgumentException("value is null"); }`

Answer: D
Section: All

Explanation/Reference:

QUESTION 38
Given:

```
1. public class Mule {
2.     public static void main(String[] args) {
3.         boolean assert = true;
4.         if(assert) {
5.             System.out.println("assert is true");
6.         }
7.     }
8. }
```

Which command-line invocations will compile?

A. `javac Mule.java`
B. `javac -source 1.3 Mule.java`
C. `javac -source 1.4 Mule.java`
D. `javac -source 1.5 Mule.java`

Answer: B
Section: All

Explanation/Reference:

QUESTION 39
Click the Exhibit button

```
1. public class A {
2.     public void method1(){
3.         B b = new B();
4.         b.method2();
5.         // more code here
6.     }
7. }
```

```
1. public class B{
2.     public void method2() {
3.         C c = new C();
4.         c.method3();
5.         // more code here
6.     }
7. }
```

```
1. public class C {
2.     public void method3(){
3.         // more code here
4.     }
5. }
```

Given:

```
25. try {
26. A a = new A();
27. a.method1();
28. } catch (Exception e) {
29. System.out.print("an error occurred");
30. }
```

Which two statements are true if a `NullPointerException` is thrown on line 3 of class `C`? (Choose two.)

A. The application will crash.
B. The code on line 29 will be executed.
C. The code on line 5 of class A will execute.
D. The code on line 5 of class B will execute.
E. The exception will be propagated back to line 27.

Answer: BE
Section: All

Explanation/Reference:

QUESTION 40
Given:

```
1. public class Venus {
2.     public static void main(String[] args) {
3.         int[] x = { 1, 2, 3 };
4.         int y[] = { 4, 5, 6 };
```

```
5.         new Venus().go(x, y);
6.     }
7.
8.     void go(int[]... z) {
9.         for (int[] a : z)
10.            System.out.print(a[0]);
11.    }
12. }
```

What is the result?

A. 1
B. 12
C. 14
D. 123
E. Compilation fails.
F. An exception is thrown at runtime.

Answer: C
Section: All

Explanation/Reference:

QUESTION 41
Given:

```
1. public class Test {
2.     public enum Dogs {collie, harrier, shepherd};
3.     public static void main(String [] args) {
4.         Dogs myDog = Dogs.shepherd;
5.         switch (myDog) {
6.         case collie:
7.             System.out.print("collie ");
8.         case default:
9.             System.out.print("retriever ");
10.        case harrier:
11.            System.out.print("harrier ");
12.        }
13.    }
14. }
```

What is the result?

A. harrier
B. shepherd
C. retriever
D. Compilation fails.
E. retriever harrier
F. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:

```
Main.java:8: illegal start of expression
```

```
        case default:
             ^
Main.java:8: illegal start of expression
        case default:
                  ^
Main.java:9: ';' expected
            System.out.print("retriever ");
                 ^
3 errors
```

QUESTION 42
Given:

```
1. public class Breaker2 {
2.     static String o = "";
3.
4.     public static void main(String[] args) {
5.         z: for (int x = 2; x < 7; x++) {
6.             if (x == 3)
7.                 continue;
8.             if (x == 5)
9.                 break z;
10.            o = o + x;
11.        }
12.        System.out.println(o);
13.    }
14. }
```

What is the result?

A. 2
B. 24
C. 234
D. 246
E. 2346
F. Compilation fails.

Answer: B
Section: All

Explanation/Reference:

QUESTION 43
Given:

```
1. public static void main(String[] args) {
2.     String str = "null";
3.     if (str == null) {
4.         System.out.println("null");
5.     } else (str.length() == 0) {
6.         System.out.println("zero");
7.     } else {
8.         System.out.println("some");
9.     }
10. }
```

What is the result?

A. `null`
B. `zero`
C. `some`
D. Compilation fails.
E. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:

```
Main.java:5: not a statement
    } else (str.length() == 0) {
           ^
Main.java:5: ';' expected
    } else (str.length() == 0) {
                           ^
Main.java:7: 'else' without 'if'
    } else {
      ^
3 errors
```

QUESTION 44
Given:

```
1. import java.io.IOException;
2.
3. class A {
4.
5.     public void process() {
6.         System.out.print("A,");
7.     }
8.
9. }
10.
11.
12.class B extends A {
13.
14.    public void process() throws IOException {
15.        super.process();
16.        System.out.print("B,");
17.        throw new IOException();
18.    }
19.
20.    public static void main(String[] args) {
21.        try {
22.            new B().process();
23.        } catch (IOException e) {
24.            System.out.println("Exception");
25.        }
26.    }
27.}
```

What is the result?

A. `Exception`
B. `A,B,Exception`
C. Compilation fails because of an error in line 20.
D. Compilation fails because of an error in line 14.
E. A `NullPointerException` is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:

```
Main.java:14: process() in B cannot override process() in A; overridden method does not throw
java.io.IOException
    public void process() throws IOException {
                ^
1 error
```

QUESTION 45
Given:

```
11. public void genNumbers() {
12.     ArrayList numbers = new ArrayList();
13.     for(int i = 0; i < 10; i++) {
14.         int value = i * ((int) Math.random());
15.         Integer intObj = new Integer(value);
16.         numbers.add(intObj);
17.     }
18.     System.out.println(numbers);
19. }
```

Which line of code marks the earliest point that an object referenced by `intObj` becomes a candidate for garbage collection?

A. Line 16
B. Line 17
C. Line 18
D. Line 19
E. The object is NOT a candidate for garbage collection.

Answer: D
Section: All

Explanation/Reference:

QUESTION 46
Given:

```
1. public class GC {
2.     private Object o;
3.     private void doSomethingElse(Object obj) { o = obj; }
4.     public void doSomething() {
5.         Object o = new Object();
6.         doSomethingElse(o);
7.         o = new Object();
```

```
8.          doSomethingElse(null);
9.          o = null;
10.     }
11. }
```

When the doSomething method is called, after which line does the `Object` created in line 5 become available for garbage collection?

A. Line 5
B. Line 6
C. Line 7
D. Line 8
E. Line 9
F. Line 10

Answer: D
Section: All

Explanation/Reference:

QUESTION 47
Given:

```
1. public class Pass2 {
2.     public void main(String[] args) {
3.         int x = 6;
4.         Pass2 p = new Pass2();
5.         p.doStuff(x);
6.         System.out.print(" main x = " + x);
7.     }
8.
9.     void doStuff(int x) {
10.        System.out.print(" doStuff x = " + x++);
11.    }
12. }
```

And the command-line invocations:

```
javac Pass2.java
java Pass2 5
```

What is the result?

A. Compilation fails.
B. An exception is thrown at runtime.
C. `doStuff x = 6 main x = 6`
D. `doStuff x = 6 main x = 7`
E. `doStuff x = 7 main x = 6`
F. `doStuff x = 7 main x = 7`

Answer: B
Section: All

Explanation/Reference:

```
The program compiled successfully, but main class was not found.
Main class should contain method: public static void main (String[] args).
```

QUESTION 48
Given:

```
1. interface DeclareStuff {
2.     public static final int EASY = 3;
3.
4.     void doStuff(int t);
5. }
6.
7. public class TestDeclare implements DeclareStuff {
8.     public static void main(String[] args) {
9.         int x = 5;
10.        new TestDeclare().doStuff(++x);
11.    }
12.
13.    void doStuff(int s) {
14.        s += EASY + ++s;
15.        System.out.println("s " + s);
16.    }
17.}
```

What is the result?

A. `s 14`
B. `s 16`
C. `s 10`
D. Compilation fails.
E. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:

```
Main.java:13: doStuff(int) in TestDeclare cannot implement doStuff(int) in DeclareStuff;
attempting to assign weaker access privileges; was public
    void doStuff(int s) {
         ^
1 error
```

QUESTION 49
A class games.cards.Poker is correctly defined in the jar file Poker.jar.
A user wants to execute the main method of Poker on a UNIX system using the command:

```
java games.cards.Poker
```

What allows the user to do this?

A. put `Poker.jar` in directory `/stuff/java`, and set the CLASSPATH to include `/stuff/java`
B. put `Poker.jar` in directory `/stuff/java`, and set the CLASSPATH to include `/stuff/java/*.jar`
C. put `Poker.jar` in directory `/stuff/java`, and set the CLASSPATH to include `/stuff/java/Poker.jar`

D. put `Poker.jar` in directory `/stuff/java/games/cards`, and set the CLASSPATH to include `/stuff/java`
E. put `Poker.jar` in directory `/stuff/java/games/cards`, and set the CLASSPATH to include `/stuff/java/*.jar`
F. put `Poker.jar` in directory `/stuff/java/games/cards`, and set the CLASSPATH to include `/stuff/java/Poker.jar`

Answer: C
Section: All

Explanation/Reference:

QUESTION 50
Given a correctly compiled class whose source code is:

```
1. package com.sun.sjcp;
2.
3. public class Commander {
4.     public static void main(String[] args) {
5.         // more code here
6.     }
7. }
```

Assume that the class file is located in `/foo/com/sun/sjcp/`, the current directory is `/foo/`, and that the classpath contains "." (current directory). Which command line correctly runs `Commander`?

A. `java Commander`
B. `java com.sun.sjcp.Commander`
C. `java com/sun/sjcp/Commander`
D. `java -cp com.sun.sjcp Commander`
E. `java -cp com/sun/sjcp Commander`

Answer: B
Section: All

Explanation/Reference:

QUESTION 51
Given:

```
interface DoStuff2 {
    float getRange(int low, int high);
}

interface DoMore {
    float getAvg(int a, int b, int c);
}

abstract class DoAbstract implements DoStuff2, DoMore {
}

06. class DoStuff implements DoStuff2 {
07.     public float getRange(int x, int y) {
08.         return 3.14f;
```

```
09.     }
10. }
11.
12. interface DoAll extends DoMore {
13.     float getAvg(int a, int b, int c, int d);
14. }
```

What is the result?

A. The file will compile without error.
B. Compilation fails. Only line 7 contains an error.
C. Compilation fails. Only line 12 contains an error.
D. Compilation fails. Only line 13 contains an error.
E. Compilation fails. Only lines 7 and 12 contain errors.
F. Compilation fails. Only lines 7 and 13 contain errors.
G. Compilation fails. Lines 7, 12, and 13 contain errors.

Answer: A
Section: All

Explanation/Reference:

QUESTION 52
Given:

```
public class Spock {
    public static void main(String[] args) {
        Long tail = 2000L;
        Long distance = 1999L;
        Long story = 1000L;
        if ((tail > distance) ^ ((story * 2) == tail))
            System.out.print("1");
        if ((distance + 1 != tail) ^ ((story * 2) == distance))
            System.out.print("2");
    }
}
```

What is the result?

A. 1
B. 2
C. 12
D. Compilation fails.
E. No output is produced.
F. An exception is thrown at runtime.

Answer: E
Section: All

Explanation/Reference:
No output since both if statements include ^ (X-OR operator). If was used an || (OR operator) instead of ^ (X-OR operator) the result will be answer A (1).

QUESTION 53
Given:

```
class Payload {
   private int weight;

   public Payload() {

   }

   public Payload(int w) {
      weight = w;
   }

   public void setWeight(int w) {
      weight = w;
   }

   public String toString() {
      return Integer.toString(weight);
   }
}

10. public class TestPayload {
11.    static void changePayload(Payload p) {
12.       /* insert code */
13.    }
14.
15.    public static void main(String[] args) {
16.       Payload p = new Payload(200);
17.       p.setWeight(1024);
18.       changePayload(p);
19.       System.out.println("p is " + p);
20.    }
21. }
```

Which code fragment, inserted at the end of line 12, produces the output `p is 420`?

A. `p.setWeight(420);`
B. `p.changePayload(420);`
C. `p = new Payload(420);`
D. `Payload.setWeight(420);`
E. `p = Payload.setWeight(420);`

Answer: A
Section: All

Explanation/Reference:
A. `p is 420`

B.
```
Main.java:23: cannot find symbol
symbol  : method changePayload(int)
location: class Payload
       p.changePayload(420);
        ^
1 error
```

C. `p is 1024`

D.
```
Main.java:23: non-static method setWeight(int) cannot be referenced from a static context
       Payload.setWeight(420);
```

```
            ^
1 error
```

E.

```
Main.java:23: non-static method setWeight(int) cannot be referenced from a static context
       p = Payload.setWeight(420);
                   ^
Main.java:23: incompatible types
found   : void
required: Payload
       p = Payload.setWeight(420);
                           ^
2 errors
```

QUESTION 54
Given:

```
class Line {
   public class Point {
      public int x, y;
   }

   public Point getPoint() {
      return new Point();
   }
}

class Triangle {
   public Triangle() {
      // insert code here
   }
}
```

Which code, inserted at line 16, correctly retrieves a local instance of a `Point` object?

A. `Point p = Line.getPoint();`
B. `Line.Point p = Line.getPoint();`
C. `Point p = (new Line()).getPoint();`
D. `Line.Point p = (new Line()).getPoint();`

Answer: D
Section: All

Explanation/Reference:
A.

```
Main.java:13: cannot find symbol
symbol  : class Point
location: class Triangle
       Point p = Line.getPoint();
       ^
Main.java:13: non-static method getPoint() cannot be referenced from a static context
       Point p = Line.getPoint();
                 ^
2 errors
```

B.

```
Main.java:13: non-static method getPoint() cannot be referenced from a static context
       Line.Point p = Line.getPoint();
                      ^
1 error
```

C.
```
Main.java:13: cannot find symbol
symbol  : class Point
location: class Triangle
       Point p = (new Line()).getPoint();
       ^
1 error
```

D.
```
The program compiled successfully
```

QUESTION 55
Given:
```
84. try {
85.     ResourceConnection con = resourceFactory.getConnection();
86.     Results r = con.query("GET INFO FROM CUSTOMER");
87.     info = r.getData();
88.     con.close();
89. } catch (ResourceException re) {
90.    errorLog.write(re.getMessage());
91. }
92. return info;
```

Which statement is true if a ResourceException is thrown on line 86?

A. Line 92 will not execute.
B. The connection will not be retrieved in line 85.
C. The resource connection will not be closed on line 88.
D. The enclosing method will throw an exception to its caller.

Answer: C
Section: All

Explanation/Reference:

QUESTION 56
Given:

```
11. class X { public void foo() { System.out.print("X "); } }
12.
13. public class SubB extends X {
14.    public void foo() throws RuntimeException {
15.        super.foo();
16.        if (true) throw new RuntimeException();
17.        System.out.print("B ");
18.    }
19.    public static void main(String[] args) {
20.        new SubB().foo();
21.    }
22. }
```

What is the result?

A. X, followed by an Exception.
B. No output, and an Exception is thrown.

C. Compilation fails due to an error on line 14.
D. Compilation fails due to an error on line 16.
E. Compilation fails due to an error on line 17.
F. `X`, followed by an Exception, followed by `B`.

Answer: A
Section: All

Explanation/Reference:

```
X Exception in thread "main" java.lang.RuntimeException
    at SubB.foo(Main.java:16)
    at SubB.main(Main.java:20)
```

QUESTION 57
Which two code fragments correctly create and initialize a static array of int elements? (Choose two.)

A. `static final int[] a = { 100,200 };`
B. `static final int[] a; static { a=new int[2]; a[0]=100; a[1]=200; }`
C. `static final int[] a = new int[2]{ 100,200 };`
D.
```
static final int[] a;
static void init() { a = new int[3]; a[0]=100; a[1]=200; }
```

Answer: AB
Section: All

Explanation/Reference:

QUESTION 58
Given:

```
class Alpha {
    public void foo() { System.out.print("Afoo "); }
}
public class Beta extends Alpha {
    public void foo() { System.out.print("Bfoo "); }
    public static void main(String[] args) {
        Alpha a = new Beta();
        Beta b = (Beta)a;
        a.foo();
        b.foo();
    }
}
```

What is the result?

A. `Afoo Afoo`
B. `Afoo Bfoo`
C. `Bfoo Afoo`
D. `Bfoo Bfoo`
E. Compilation fails.
F. An exception is thrown at runtime.

Answer: D
Section: All

Explanation/Reference:

```
Bfoo Bfoo
```

QUESTION 59
Which two scenarios are NOT safe to replace a `StringBuffer` object with a `StringBuilder` object? (Choose two.)

A. When using versions of Java technology earlier than 5.0.
B. When sharing a `StringBuffer` among multiple threads.
C. When using the `java.io` class `StringBufferInputStream`.
D. When you plan to reuse the `StringBuffer` to build more than one string.

Answer: AB
Section: All

Explanation/Reference: