Oracle

Exam Questions 1Z0-803

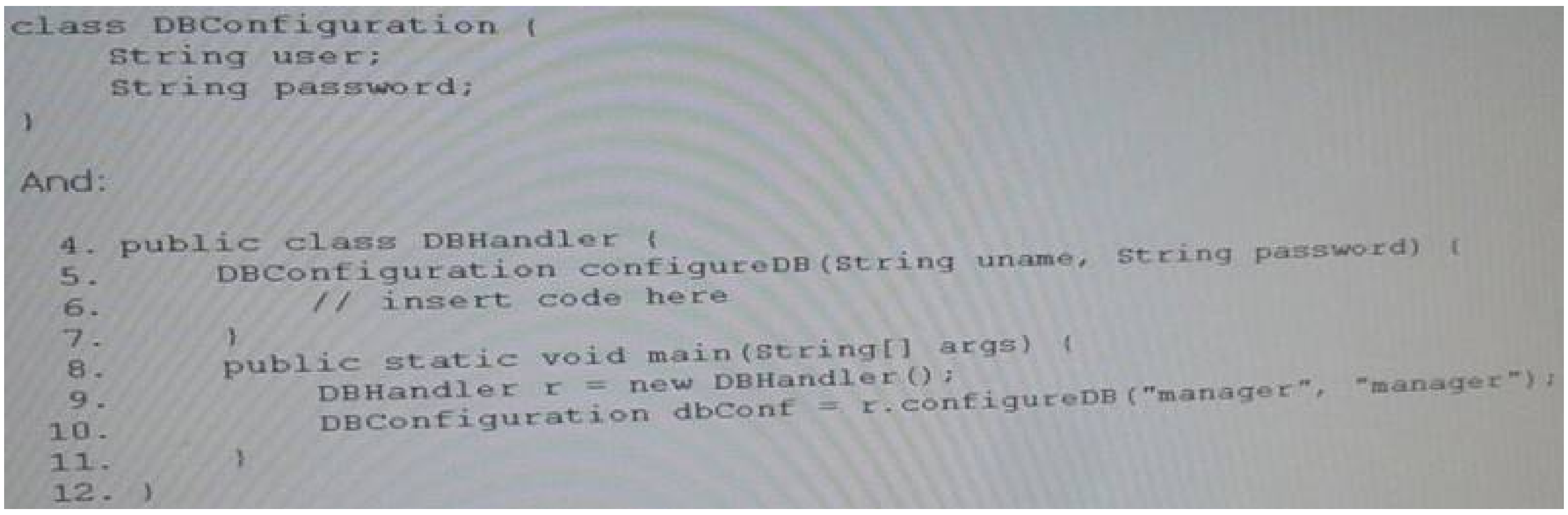
Java SE 7 Programmer I

NEW QUESTION 1
Which code fragment cause a compilation error?

A. flat flt = 100F;
B. float flt = (float) 1_11.00;
C. float flt = 100;
D. double y1 = 203.22; floatflt = y1
E. int y2 = 100; floatflt = (float) y2;

Answer: B

NEW QUESTION 2
Given the code format:



```
class DBConfiguration {
     String user;
     String password;
}
```

And:

```
 4. public class DBHandler {
 5.       DBConfiguration configureDB(String uname, String password) {
 6.            // insert code here
 7.       }
 8.       public static void main(String[] args) {
 9.            DBHandler r = new DBHandler();
10.            DBConfiguration dbConf = r.configureDB("manager", "manager");
11.       }
12. }
```

Which code fragment must be inserted at line 6 to enable the code to compile?

A. DBConfiguration f; return f;
B. Return DBConfiguration;
C. Return new DBConfiguration;
D. Retutn 0;

Answer: B

NEW QUESTION 3
Given:
public class SampleClass {
public static void main(String[] args) {
AnotherSampleClass asc = new AnotherSampleClass(); SampleClass sc = new SampleClass();
sc = asc;
System.out.println("sc: " + sc.getClass()); System.out.println("asc: " + asc.getClass());
}}
class AnotherSampleClass extends SampleClass {
}
What is the result?

A. sc: class Objectasc: class AnotherSampleClass
B. sc: class SampleClassasc: class AnotherSampleClass
C. sc: class AnotherSampleClass asc: class SampleClass
D. sc: class AnotherSampleClassasc: class AnotherSampleClass

Answer: D

NEW QUESTION 4
Given the code fragment:
System.out.println(2 + 4 * 9 - 3); //Line 21
System.out.println((2 + 4) * 9 - 3); // Line 22
System.out.println(2 + (4 * 9) - 3); // Line 23
System.out.println(2 + 4 * (9 - 3)); // Line 24
System.out.println((2 + 4 * 9) - 3); // Line 25 Which line of codes prints the highest number?

A. Line 21
B. Line 22
C. Line 23
D. Line 24
E. Line 25

Answer: B

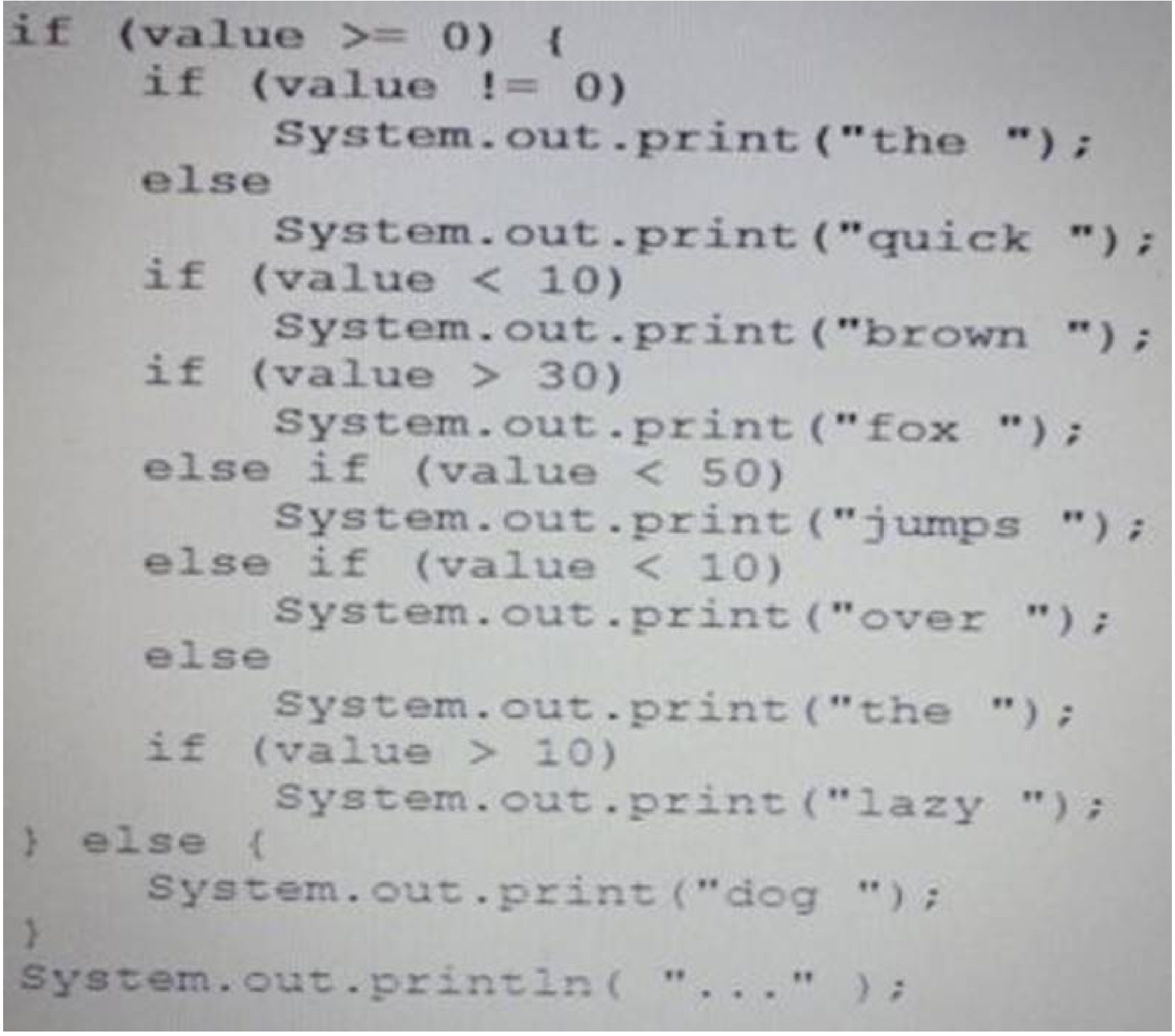
Explanation: The following is printed: 35
51
35
26
35

NEW QUESTION 5
Given the code fragment:
public static void main(String[] args) {
int iArray[] = {65, 68, 69};
iArray[2] = iArray[0];
iArray[0] = iArray[1];
iArray[1] = iArray[2];
for (int element : iArray) {
System.out.print(element + " ");
}

A. 68, 65, 69
B. 68, 65, 65
C. 65,68, 65
D. 65, 68, 69
E. Compilation fails

Answer: B

NEW QUESTION 6
Given the following code fragment:



```
if (value >= 0) {
    if (value != 0)
        System.out.print("the ");
    else
        System.out.print("quick ");
    if (value < 10)
        System.out.print("brown ");
    if (value > 30)
        System.out.print("fox ");
    else if (value < 50)
        System.out.print("jumps ");
    else if (value < 10)
        System.out.print("over ");
    else
        System.out.print("the ");
    if (value > 10)
        System.out.print("lazy ");
} else {
    System.out.print("dog ");
}
System.out.println( "..." );
```

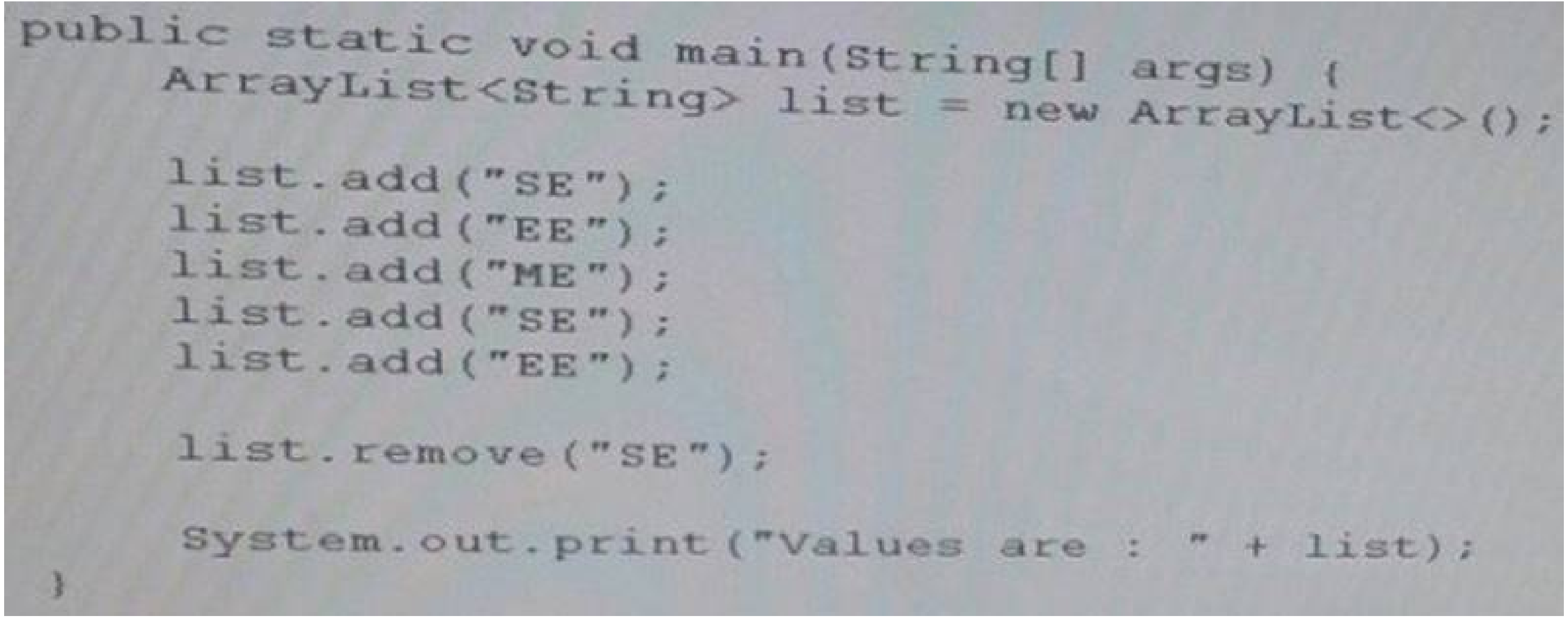
What is the result if the integer value is 33?

A. The fox jump lazy …
B. The fox lazy …
C. Quick fox over lazy …
D. Quick fox the ….

Answer: B

Explanation: 33 is greater than 0.
33 is not equal to 0. the is printed.
33 is greater than 30 fox is printed
33 is greater then 10 (the two else if are skipped) lazy is printed
finally … is printed.

NEW QUESTION 7
Given the code fragment:



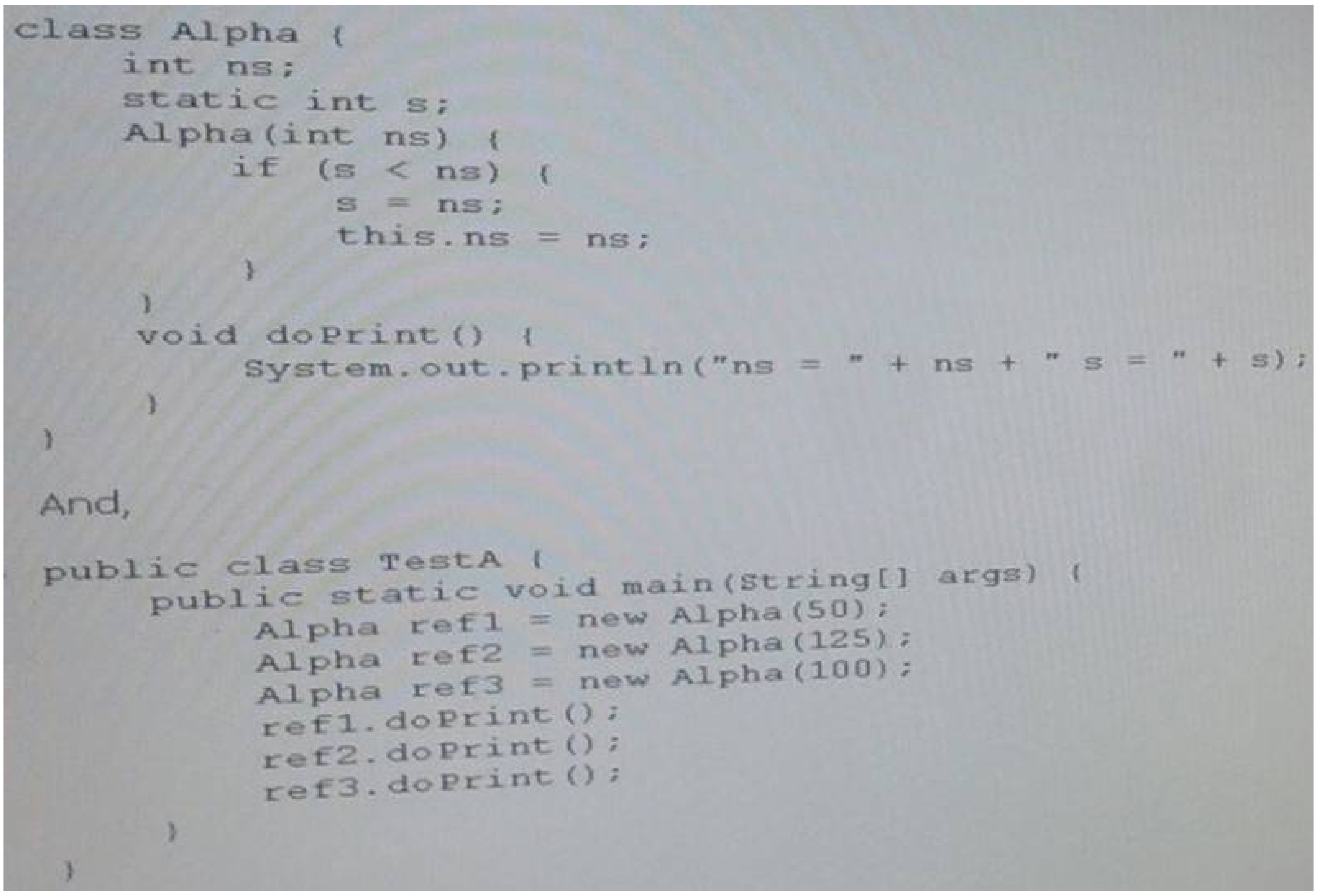
```
public static void main(String[] args) {
    ArrayList<String> list = new ArrayList<>();

    list.add("SE");
    list.add("EE");
    list.add("ME");
    list.add("SE");
    list.add("EE");

    list.remove("SE");

    System.out.print("Values are : " + list);
}
```

What is the result?

A. Values are : [EE, ME]
B. Values are : [EE, EE, ME]
C. Values are : [EE, ME, EE]
D. Values are : [SE, EE, ME, EE]
E. Values are : [EE, ME, SE, EE]

Answer: E

NEW QUESTION 8
Given:



```
class Alpha {
    int ns;
    static int s;
    Alpha(int ns) {
        if (s < ns) {
            s = ns;
            this.ns = ns;
        }
    }
    void doPrint() {
        System.out.println("ns = " + ns + " s = " + s);
    }
}
```

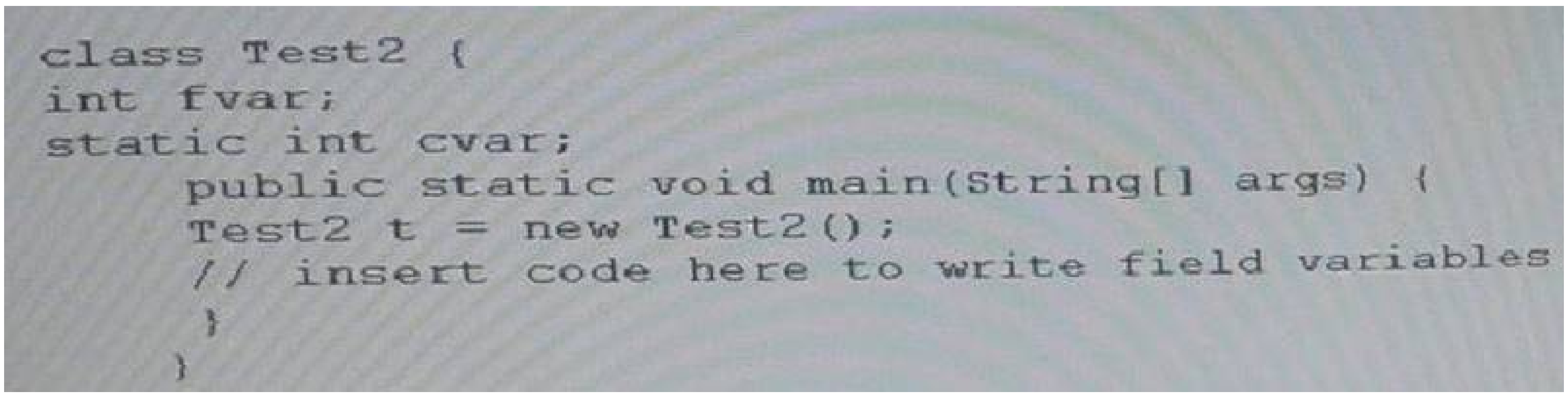
And,

```
public class TestA {
    public static void main(String[] args) {
        Alpha ref1 = new Alpha(50);
        Alpha ref2 = new Alpha(125);
        Alpha ref3 = new Alpha(100);
        ref1.doPrint();
        ref2.doPrint();
        ref3.doPrint();
    }
}
```

A. ns = 50 S = 125ns = 125 S = 125ns = 100 S = 125
B. ns = 50 S = 125 ns = 125 S = 125ns = 0 S = 125
C. ns = 50 S = 50 ns = 125 S = 125ns = 100 S = 100
D. ns = 50 S = 50 ns = 125 S = 125ns = 0 S = 125

Answer: B

NEW QUESTION 9
Given the code fragment



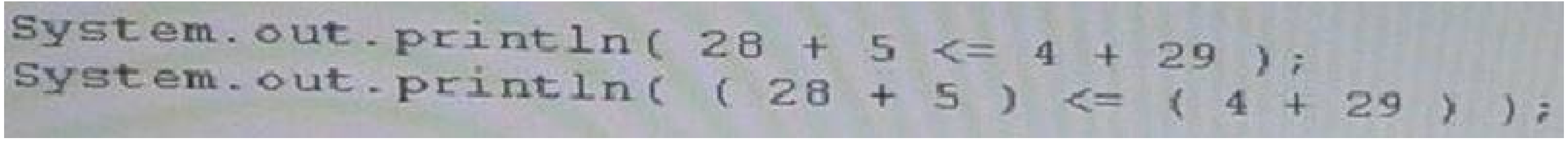
```
class Test2 {
int fvar;
static int cvar;
    public static void main(String[] args) {
    Test2 t = new Test2();
    // insert code here to write field variables
    }
  }
```

Which code fragments, inserted independently, enable the code compile?

A. t.fvar = 200;
B. cvar = 400;
C. fvar = 200; cvar = 400;
D. this.fvar = 200; this.cvar = 400;
E. t.fvar = 200; Test2.cvar = 400;
F. this.fvar = 200; Test2.cvar = 400;

Answer: B

NEW QUESTION 10
Given the code fragment:



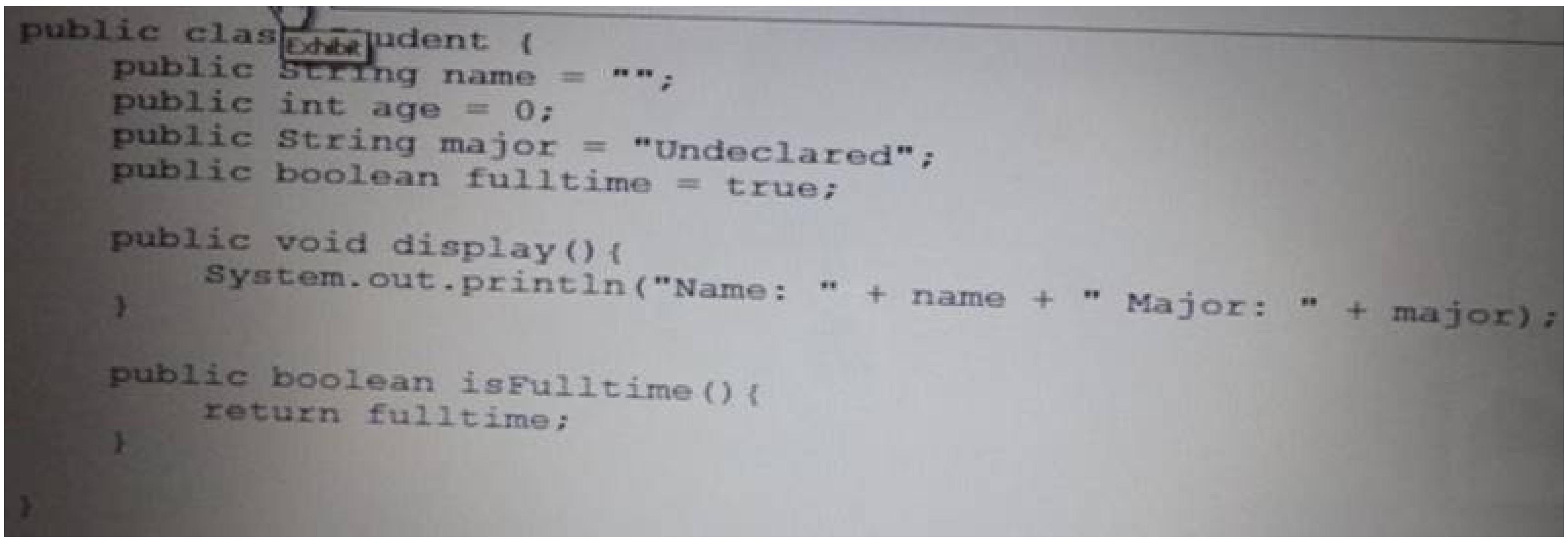
```
System.out.println( 28 + 5 <= 4 + 29 );
System.out.println( ( 28 + 5 ) <= ( 4 + 29 ) );
```

What is the result?

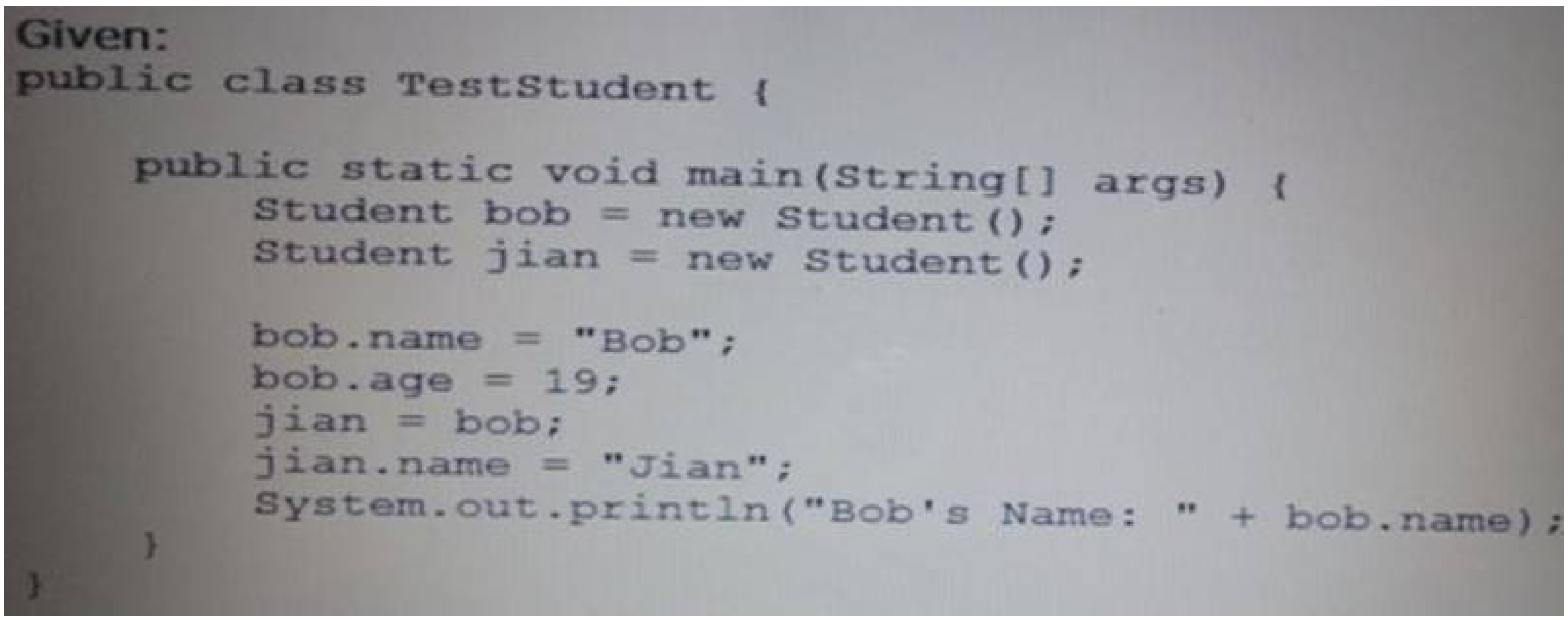
A. 28false29 true
B. 285 < 429 true
C. true true
D. compilation fails

Answer: C

NEW QUESTION 10
View the exhibit:



```
public class Student {
    public String name = "";
    public int age = 0;
    public String major = "Undeclared";
    public boolean fulltime = true;

    public void display(){
        System.out.println("Name: " + name + " Major: " + major);
    }

    public boolean isFulltime(){
        return fulltime;
    }
}
```



```
Given:
public class TestStudent {

    public static void main(String[] args) {
        Student bob = new Student();
        Student jian = new Student();

        bob.name = "Bob";
        bob.age = 19;
        jian = bob;
        jian.name = "Jian";
        System.out.println("Bob's Name: " + bob.name);
    }
}
```

What is the result when this program is executed?

A. Bob's Name: Bob
B. Bob's Name: Jian

C. Nothing prints
D. Bob’s name

Answer: B

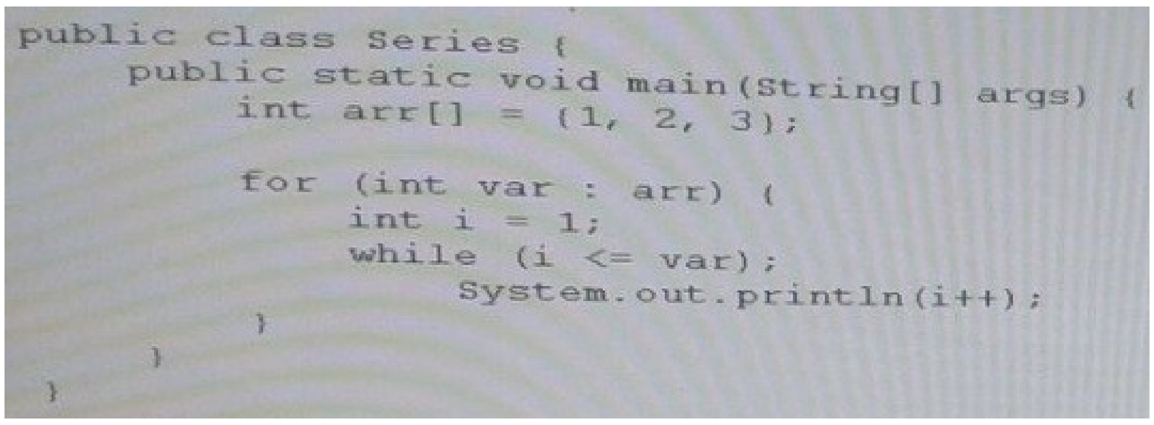
Explanation: After the statement jian = bob; the jian will reference the same object as bob.

NEW QUESTION 13
What is the proper way to defined a method that take two int values and returns their sum as an int value?

A. int sum(int first, int second) { first + second; }
B. int sum(int first, second) {return first + second; }
C. sum(int first, int second) { return first + second; }
D. int sum(int first, int second) { return first + second; }
E. void sum (int first, int second) { return first + second; }

Answer: D

NEW QUESTION 17
Given:



```
public class Series {
    public static void main(String[] args) {
        int arr[] = {1, 2, 3};

        for (int var : arr) {
            int i = 1;
            while (i <= var);
                System.out.println(i++);
        }
    }
}
```

What is the result?

A. 111
B. 123
C. 234
D. Compilation fails
E. The loop executes infinite times

Answer: E

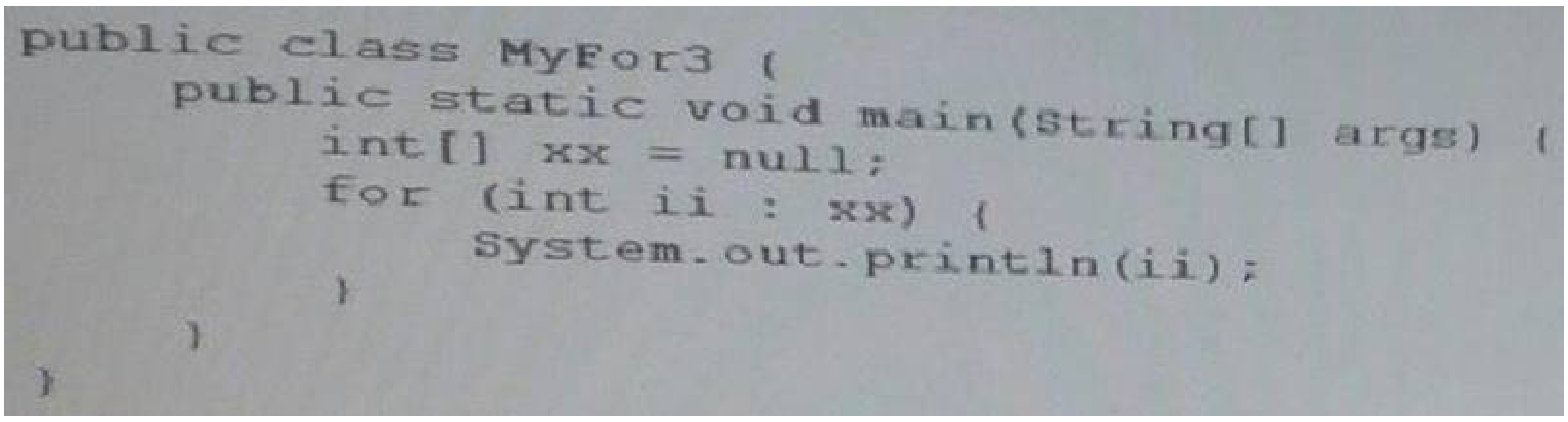
NEW QUESTION 22
Given:
import java.util.*; public class Ref {
public static void main(String[] args) {
StringBuilder s1 = new StringBuilder("Hello Java!"); String s2 = s1.toString();
List<String> lst = new ArrayList<String>(); lst.add(s2); System.out.println(s1.getClass()); System.out.println(s2.getClass()); System.out.println(lst.getClass());
}
}
What is the result?

A. class java.lang.String class java.lang.String class java.util.ArrayList
B. class java.lang.Object class java.lan
C. Object classjava.util.Collection
D. class java.lang.StringBuilder class java.lang.Stringclass java.util.ArrayList
E. class java.lang.StringBuilder class java.lang.Stringclass java.util.List

Answer: C

Explanation: class java.lang.StringBuilder class java.lang.String
class java.util.ArrayList

NEW QUESTION 27
Given:



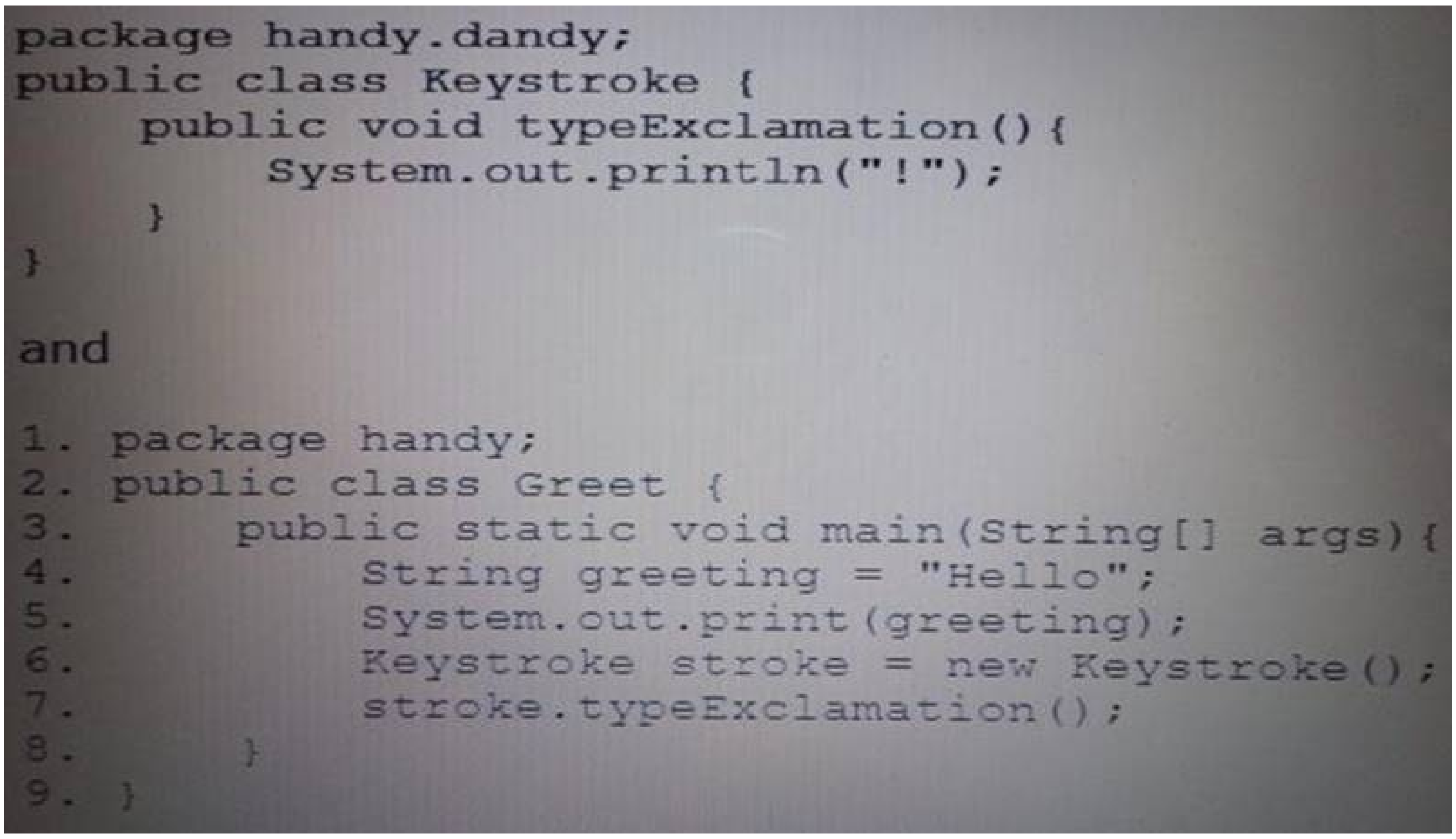
```
public class MyFor3 {
    public static void main(String[] args) {
        int[] xx = null;
        for (int ii : xx) {
            System.out.println(ii);
        }
    }
}
```

What is the result?

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

Answer: C

NEW QUESTION 30
Given:



```
package handy.dandy;
public class Keystroke {
    public void typeExclamation(){
        System.out.println("!");
    }
}
```

and

```
1. package handy;
2. public class Greet {
3.     public static void main(String[] args){
4.         String greeting = "Hello";
5.         System.out.print(greeting);
6.         Keystroke stroke = new Keystroke();
7.         stroke.typeExclamation();
8.     }
9. }
```

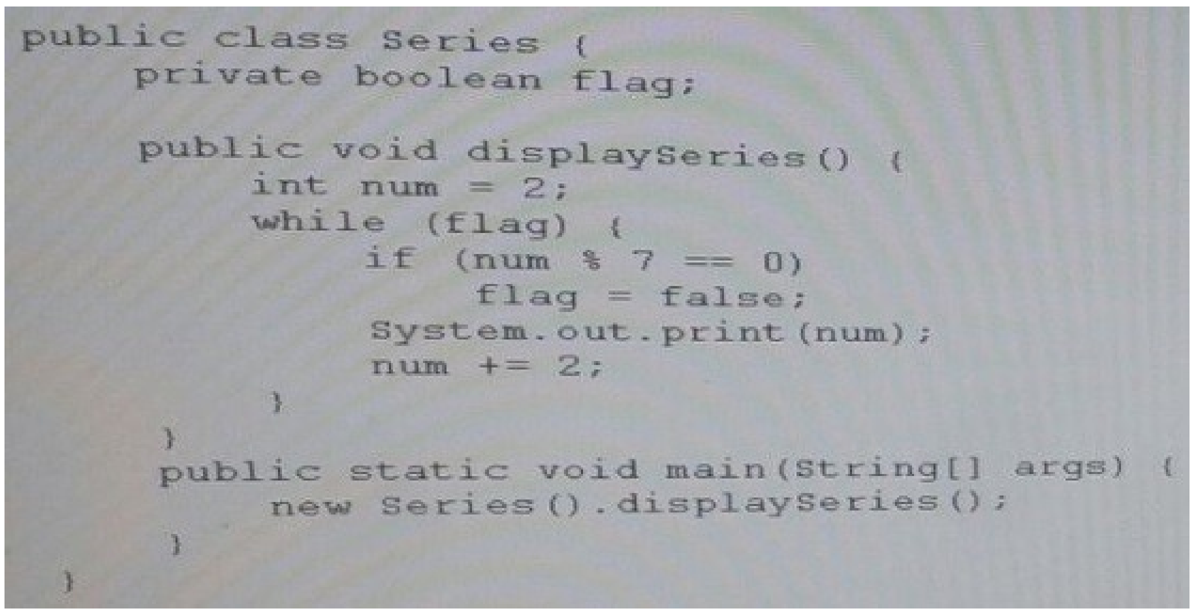
What three modifications, made independently, made to class greet, enable the code to compile and run?

A. line 6 replaced with handy.dandy.keystroke stroke = new KeyStroke ( );
B. line 6 replaced with handy.*.KeyStroke = new KeyStroke ( );
C. line 6 replaced with handy.dandy.KeyStroke Stroke = new handy.dandy.KeyStroke();
D. import handy.*; added before line 1
E. import handy.dandy.*; added after line 1
F. import handy.dandy,KeyStroke; added after line 1
G. import handy.dandy.KeyStroke.typeException(); added before line 1

Answer: CEF

Explanation: Three separate solutions:
C: the full class path to the method must be stated (when we have not imported the package)
D: We can import the hold dandy class F: we can import the specific method

NEW QUESTION 33
Given:



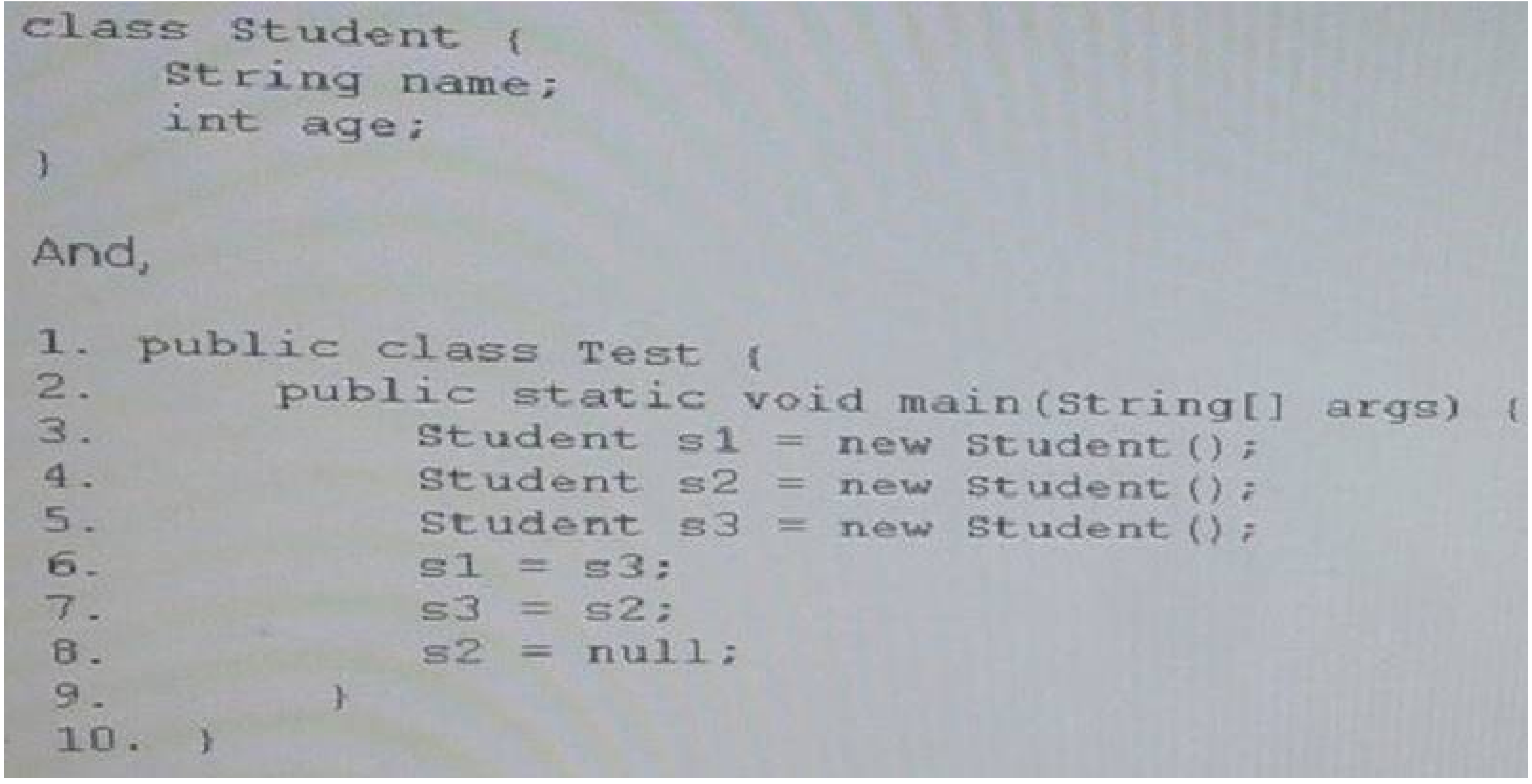
```
public class Series {
    private boolean flag;

    public void displaySeries() {
        int num = 2;
        while (flag) {
            if (num % 7 == 0)
                flag = false;
            System.out.print(num);
            num += 2;
        }
    }
    public static void main(String[] args) {
        new Series().displaySeries();
    }
}
```

What is the result?

A. 2 4 6 8 10 12
B. 2 4 6 8 10 12 14
C. Compilation fails
D. The program prints multiple of 2 infinite times
E. The program prints nothing

Answer: B

NEW QUESTION 38
Given the code fragment:



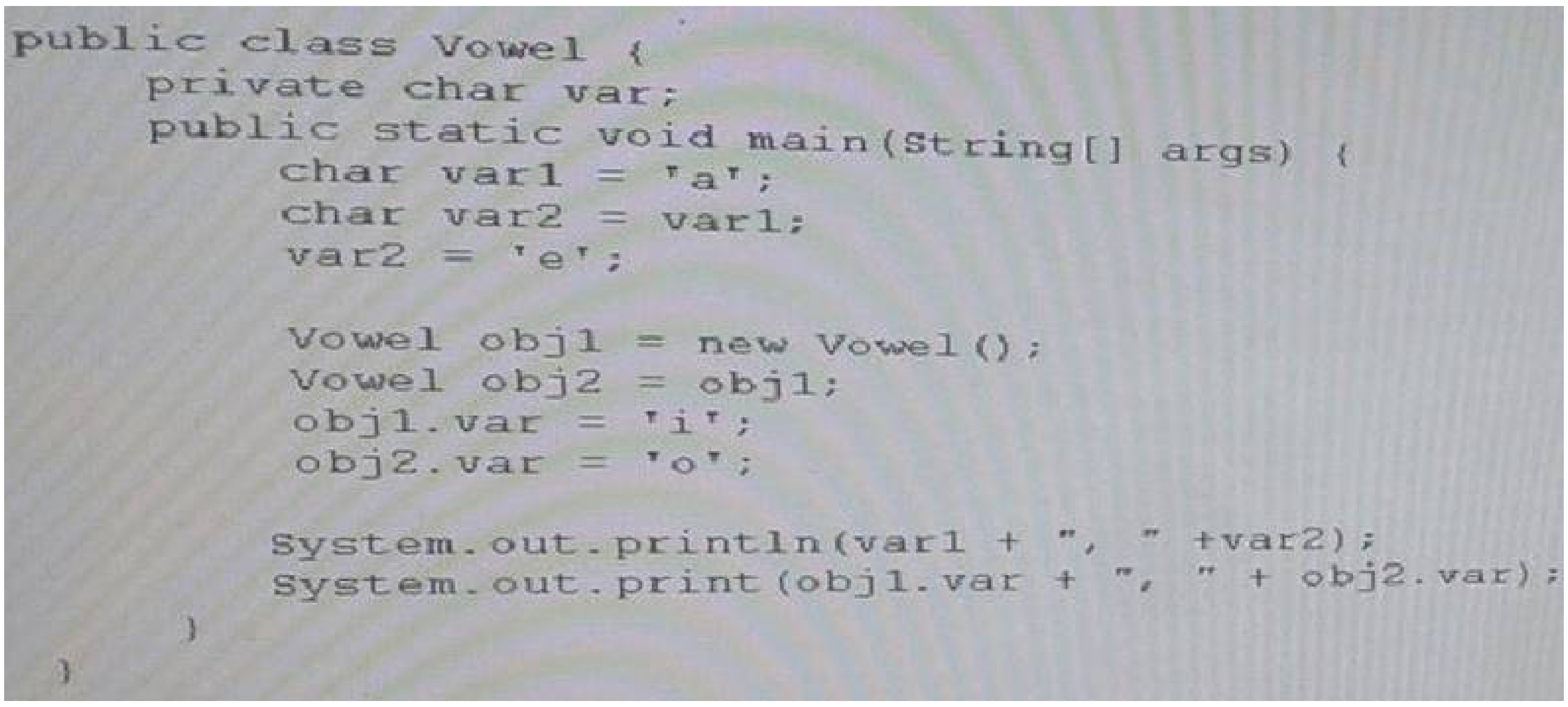
```
class Student {
    String name;
    int age;
}

And,

1.  public class Test {
2.      public static void main(String[] args) {
3.          Student s1 = new Student();
4.          Student s2 = new Student();
5.          Student s3 = new Student();
6.          s1 = s3;
7.          s3 = s2;
8.          s2 = null;
9.      }
10. }
```

Which statement is true?

A. After line 8, three objects are eligible for garbage collection
B. After line 8, two objects are eligible for garbage collection
C. After line 8, one object is eligible for garbage collection
D. After line 8, none of the objects are eligible for garbage collection

Answer: C

NEW QUESTION 43
Given:

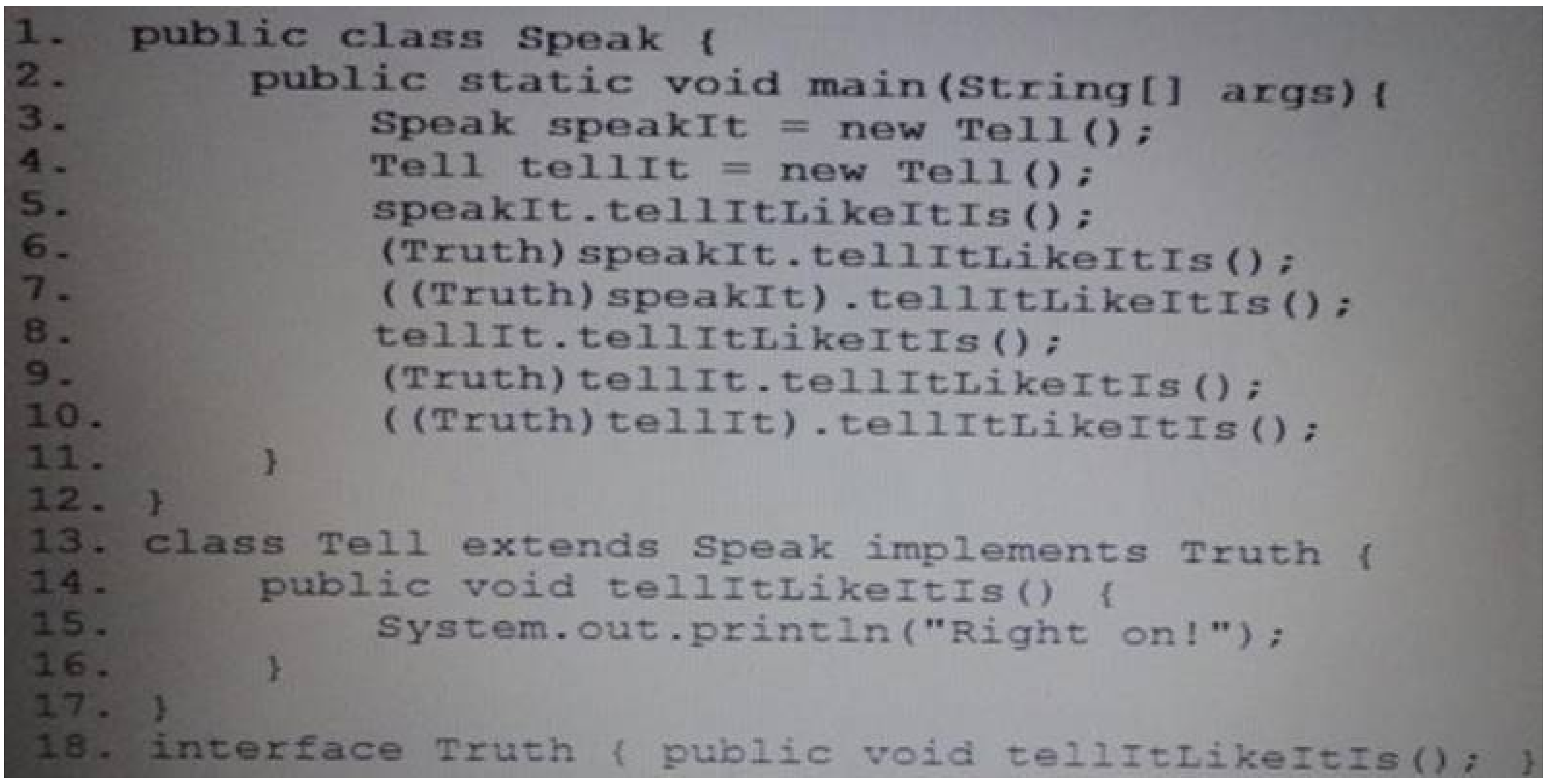


```
public class Vowel {
    private char var;
    public static void main(String[] args) {
        char var1 = 'a';
        char var2 = var1;
        var2 = 'e';

        Vowel obj1 = new Vowel();
        Vowel obj2 = obj1;
        obj1.var = 'i';
        obj2.var = 'o';

       System.out.println(var1 + ", " +var2);
       System.out.print(obj1.var + ", " + obj2.var);
    }
}
```

A. a, e i, o
B. a, e o, o
C. e, e I, o
D. e, e o, o

Answer: B

NEW QUESTION 44
Given:



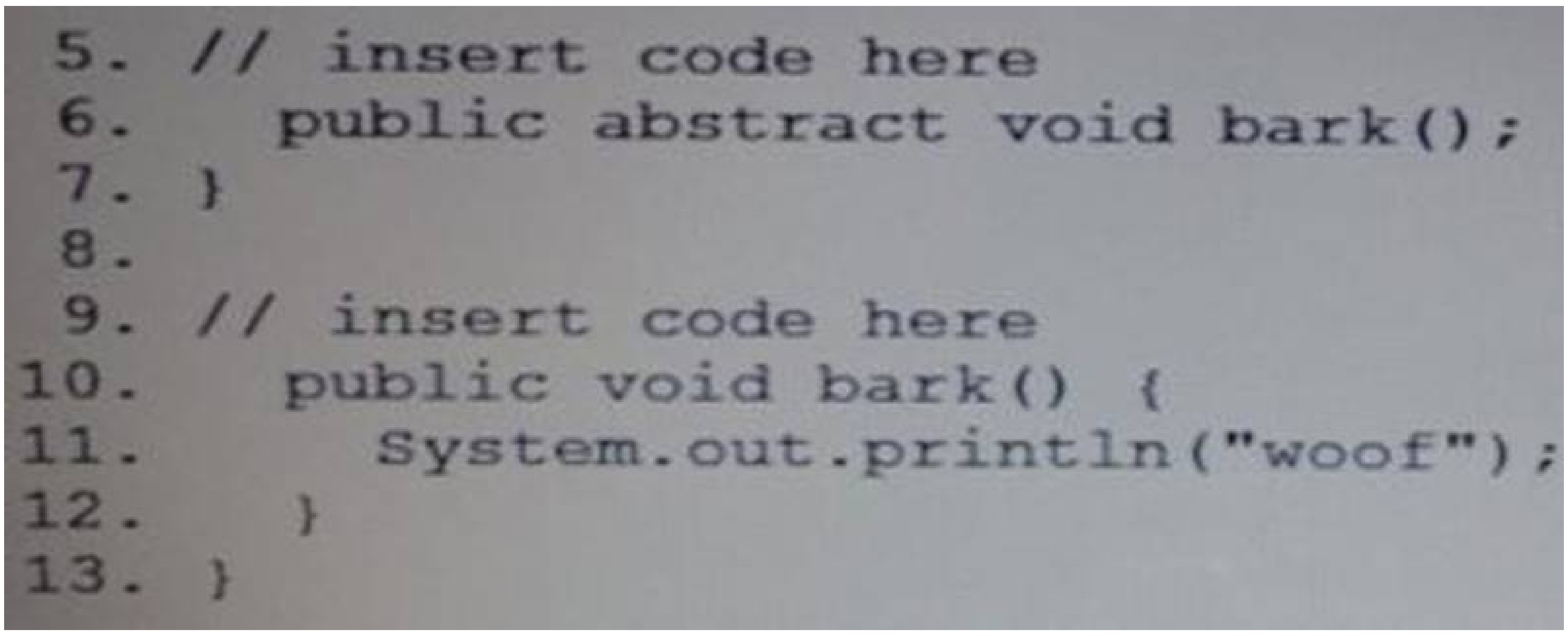
```
1.  public class Speak {
2.      public static void main(String[] args){
3.          Speak speakIt = new Tell();
4.          Tell tellIt = new Tell();
5.          speakIt.tellItLikeItIs();
6.          (Truth)speakIt.tellItLikeItIs();
7.          ((Truth)speakIt).tellItLikeItIs();
8.          tellIt.tellItLikeItIs();
9.          (Truth)tellIt.tellItLikeItIs();
10.         ((Truth)tellIt).tellItLikeItIs();
11.     }
12. }
13. class Tell extends Speak implements Truth {
14.     public void tellItLikeItIs() {
15.         System.out.println("Right on!");
16.     }
17. }
18. interface Truth { public void tellItLikeItIs(); }
```

Which three lines will compile and output “right on!”?

A. Line 5
B. Line 6
C. Line 7
D. Line 8
E. Line 9
F. Line 10

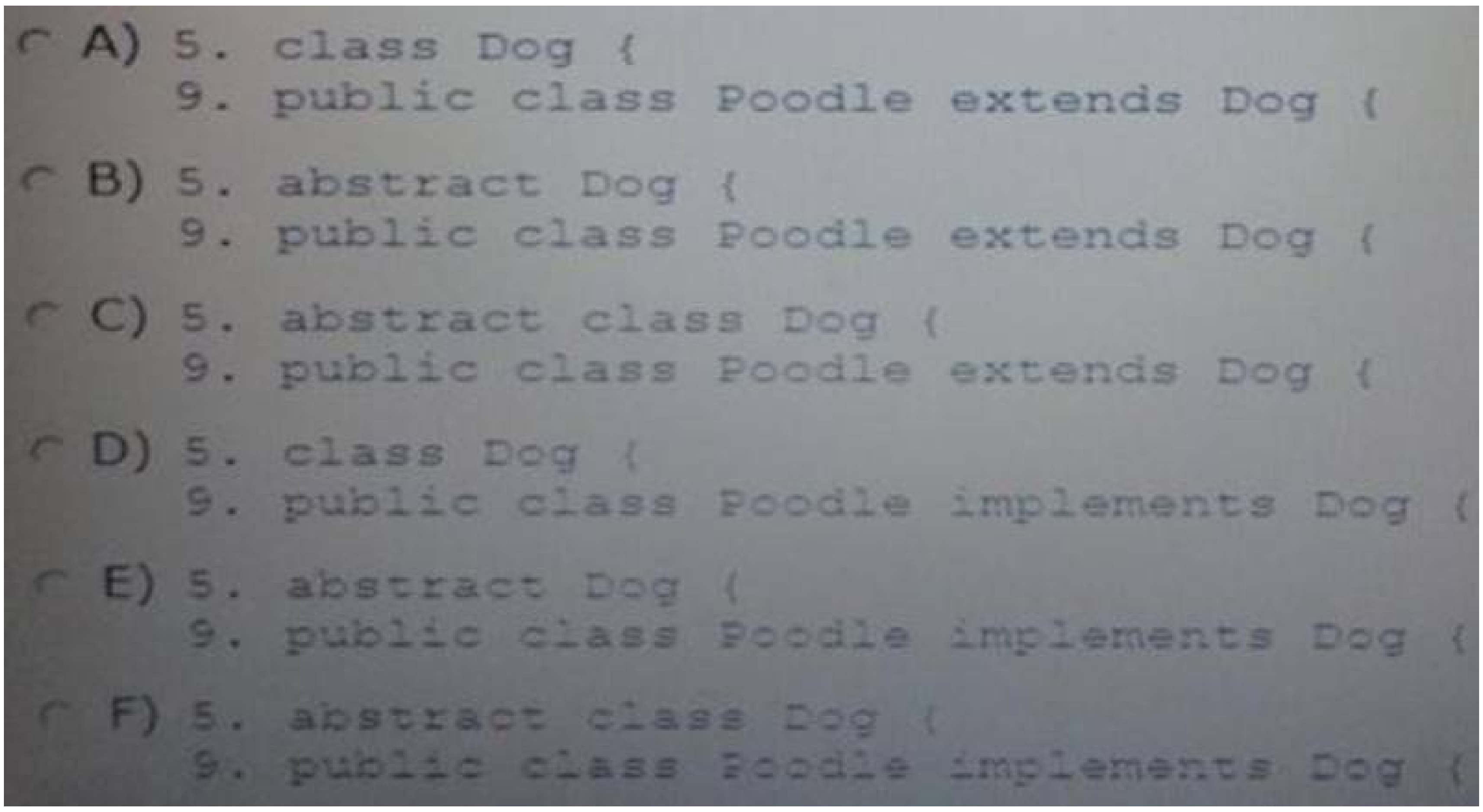
Answer: CDF

NEW QUESTION 45
Given:



```
 5. // insert code here
 6.    public abstract void bark();
 7. }
 8.
 9. // insert code here
10.    public void bark() {
11.       System.out.println("woof");
12.    }
13. }
```

What code should be inserted?



```
○ A) 5. class Dog {
     9. public class Poodle extends Dog {

○ B) 5. abstract Dog {
     9. public class Poodle extends Dog {

○ C) 5. abstract class Dog {
     9. public class Poodle extends Dog {

○ D) 5. class Dog {
     9. public class Poodle implements Dog {

○ E) 5. abstract Dog {
     9. public class Poodle implements Dog {

○ F) 5. abstract class Dog {
     9. public class Poodle implements Dog {
```

A. Option A
B. Option B
C. Option C
D. Option D
E. Option E
F. Option F

Answer: C

Explanation: Dog should be an abstract class. The correct syntax for this is: abstract class Dog { Poodle should extend Dog (not implement).

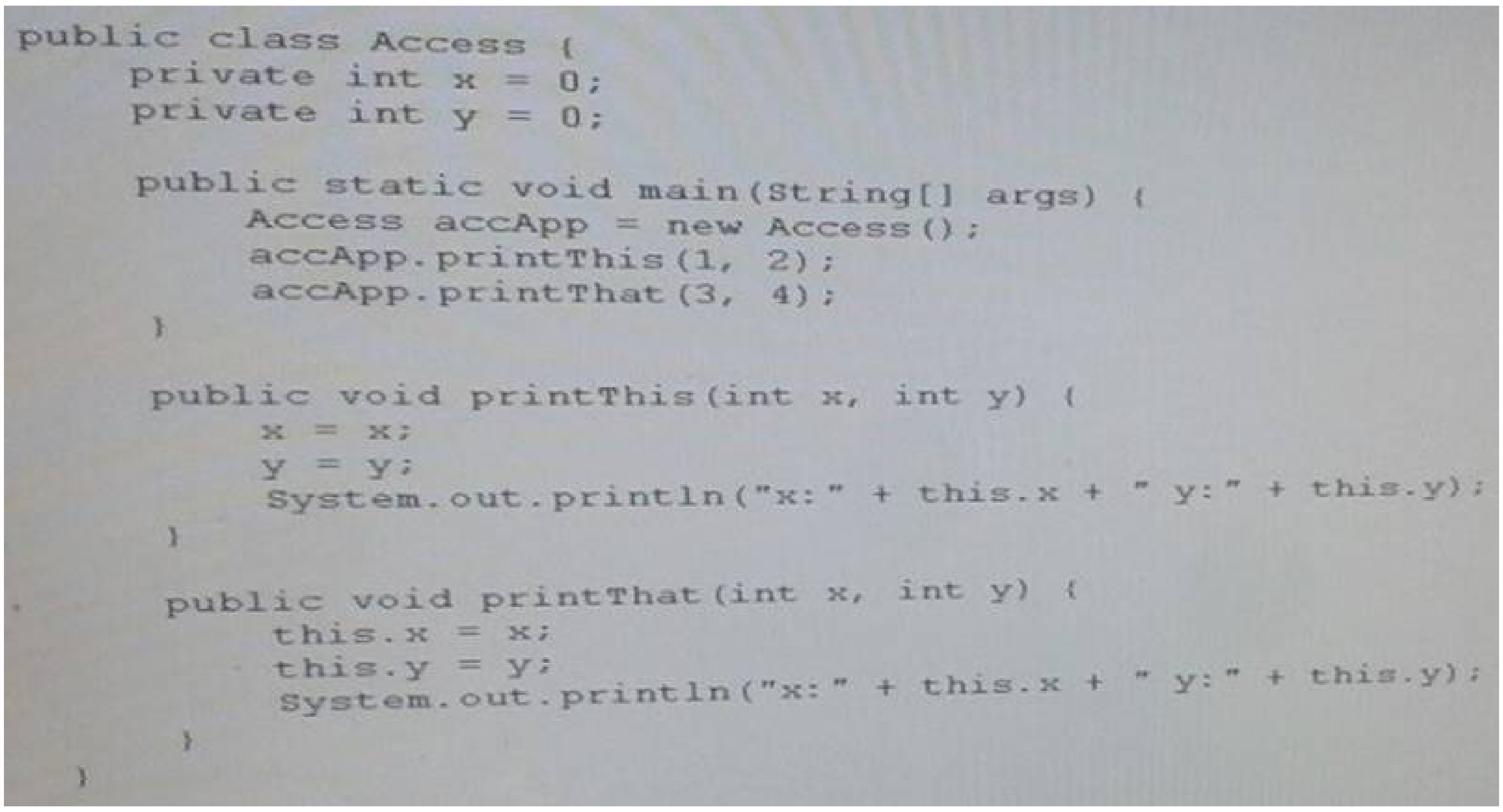
NEW QUESTION 46
Given:
public class TestField { int x;
int y;
public void doStuff(int x, int y) { this.x = x;
y =this.y;
}
public void display() {
System.out.print(x + " " + y + " : ");
}
public static void main(String[] args) { TestField m1 = new TestField(); m1.x = 100;
m1.y = 200;
TestField m2 = new TestField(); m2.doStuff(m1.x, m1.y); m1.display();
m2.display();
}
}
What is the result?

A. 100 200 : 100 200
B. 100 0 : 100 0 :

C. 100 200 : 100 0 :
D. 100 0 : 100 200 :

Answer: C

NEW QUESTION 50
Given:



```
public class Access {
    private int x = 0;
    private int y = 0;

    public static void main(String[] args) {
        Access accApp = new Access();
        accApp.printThis(1, 2);
        accApp.printThat(3, 4);
    }

    public void printThis(int x, int y) {
        x = x;
        y = y;
        System.out.println("x:" + this.x + " y:" + this.y);
    }

    public void printThat(int x, int y) {
        this.x = x;
        this.y = y;
        System.out.println("x:" + this.x + " y:" + this.y);
    }
}
```

What is the result?

A. x: 1 y: 2
B. 3 y: 4
C. x: 0 y: 0
D. 3 y: 4
E. x: 1 y: 2
F. 0 y: 0
G. x: 0 y: 0
H. 0 y: 0

Answer: C

NEW QUESTION 51
Give:
Public Class Test {
}
Which two packages are automatically imported into the java source file by the java compiler?

A. Java.lang
B. Java.awt
C. Java.util
D. Javax.net
E. Java.*
F. The package with no name

Answer: AF

Explanation: For convenience, the Java compiler automatically imports three entire packages for each source file: (1) the package with no name, (2) the java.lang package, and (3) the current package (the package for the current file).
Note:Packages in the Java language itself begin withjava.orjavax.

NEW QUESTION 54
The catch clause argument is always of type ________.

A. Exception
B. Exception but NOT including RuntimeException
C. Throwable
D. RuntimeException
E. CheckedException
F. Error

Answer: C

Explanation: Because all exceptions in Java are the sub-class ofjava.lang.Exceptionclass, you can have a singlecatch blockthat catches an exception of

typeExceptiononly. Hence the compiler is fooled into thinking that this block canhandle any exception.
See the following example:

```
try
{
// ...
}
catch(Exception ex)
{
// Exception handling code for ANY exception
}
```

You can also use the java.lang.Throwable class here, since Throwable is the parent class for the application-specificException classes. However, this is discouraged in Java programming circles. This is because Throwable happens to also be the parent class for the non-application specific Error classes which are not meant to be handled explicitly as they are catered forby the JVM itself.
Note: The Throwable class is the superclass of all errors and exceptions in the Java language. Only objects that are instances of this class (or one of its subclasses) are thrown by the Java Virtual Machine or can be thrown by the Java throw statement.
A throwable contains a snapshot of the execution stack of its thread at the time it was created. It can also contain a message string that gives more information about the error.

NEW QUESTION 58
Which two statements are true for a two-dimensional array?

A. It is implemented as an array of the specified element type.
B. Using a row by column convention, each row of a two-dimensional array must be of the same size.
C. At declarationtime, the number of elements of the array in each dimension must be specified.
D. All methods of the class Object may be invoked on the two-dimensional array.

Answer: AD

NEW QUESTION 61
Which two statements are true for a two-dimensional array of primitive data type?

A. It cannot contain elements of different types.
B. The length of each dimension must be the same.
C. At the declaration time, the number of elements of the array in each dimension must be specified.
D. All methods of the class object may be invoked on the two-dimensional array.

Answer: CD

Explanation: http://stackoverflow.com/questions/12806739/is-an-array-a-primitive-type-or-an-object-or-something-else-entirely

NEW QUESTION 64
Which two statements correctly describe checked exception?

A. These are exceptional conditions that a well-written application should anticipate and recover from.
B. These are exceptional conditions that are external to the application, and that the application usually cannot anticipate or recover from.
C. These are exceptional conditions that are internal to the application, and that the application usually cannot anticipate or recover from.
D. Every class that is a subclass of RuntimeException and Error is categorized as checked exception.
E. Every class that is a subclass of Exception, excluding RuntimeException and its subclasses, is categorized as checked exception.

Answer: AE

Explanation: Reference: Checked versus unchecked exceptions

NEW QUESTION 67
Given:

```
public class App {
// Insert code here
System.out.print("Welcome to the world of Java");
}
}
```

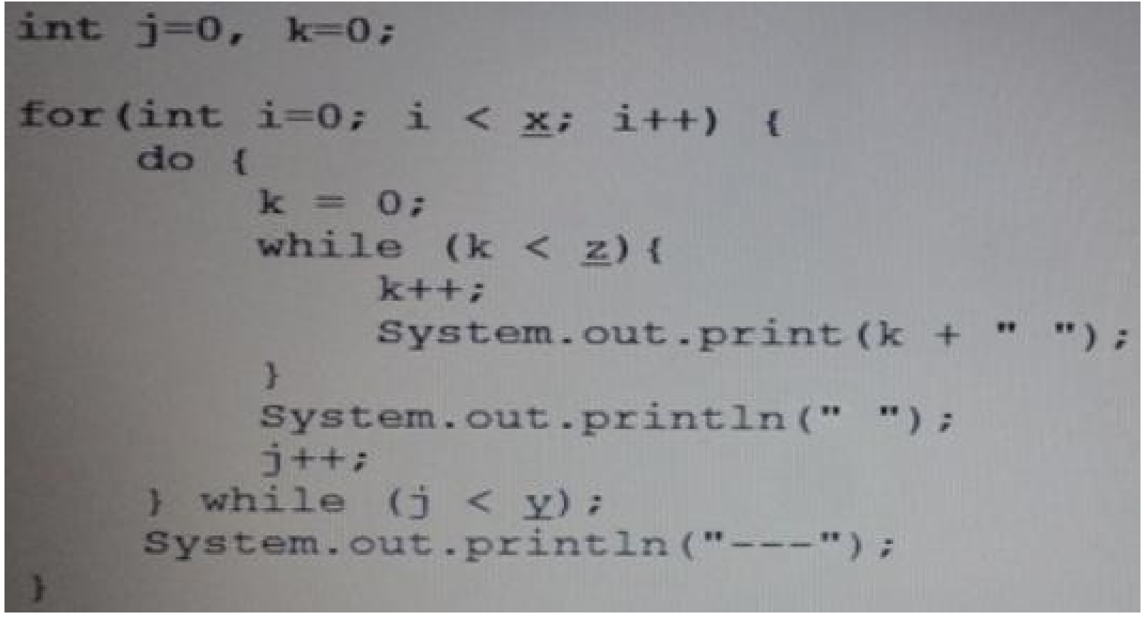
Which two code fragments, when inserted independently at line // Insert code here, enable the program to execute and print the welcome message on the screen?

A. static public void main (String [] args) {
B. static void main (String [] args) {
C. public static void Main (String [] args) {
D. public staticvoid main (String [] args) {
E. public void main (String [] args) {

Answer: AD

Explanation: Incorrect:
Not B: No main class found.
Not C: Main method not found not E: Main method is not static.

NEW QUESTION 71
Given the code fragment:



```
int j=0, k=0;

for(int i=0; i < x; i++) {
      do {
            k = 0;
            while (k < z){
                  k++;
                  System.out.print(k + " ");
            }
            System.out.println(" ");
            j++;
      } while (j < y);
      System.out.println("---");
}
```

What values ofx,y,zwill produce the following result?
1 2 3 4
1 2 3 4
1 2 3 4
----
1 2 3 4

A. X = 4, Y = 3, Z= 2
B. X = 3, Y = 2, Z = 3
C. X = 2, Y = 3, Z = 3
D. X = 4, Y = 2, Z = 3
E. X = 2, Y = 3, Z = 4

Answer: E

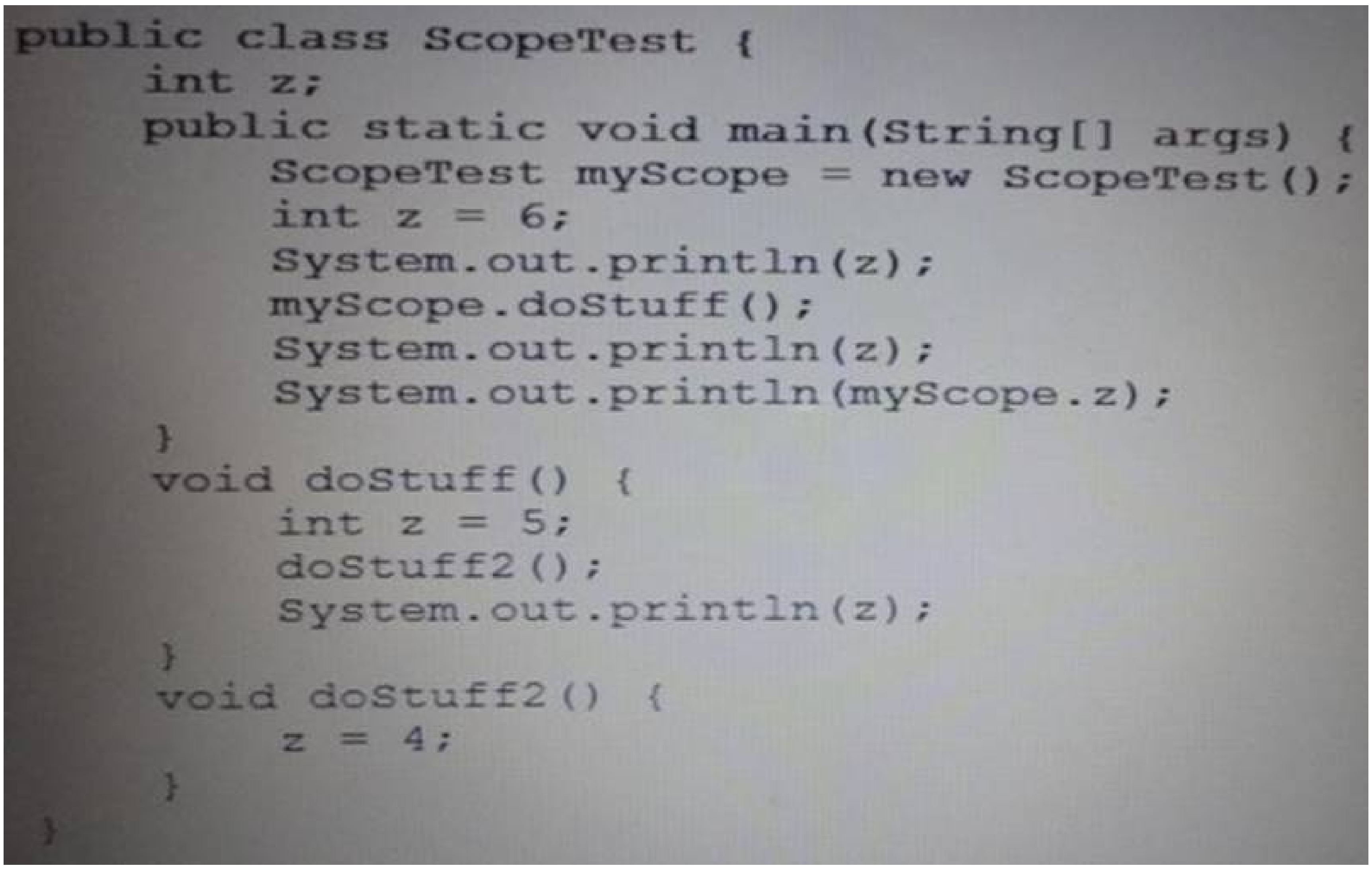
Explanation: Z is for the innermost loop. Should print 1 2 3 4. So Z must be 4.
Y is for the middle loop. Should print three lines of 1 2 3 4. So Ymust be set 3. X is for the outmost loop. Should print 2 lines of. So X should be 2.

NEW QUESTION 73
public class ForTest {
public static void main(String[] args) { int[] arrar= {1,2,3};
for (foo) {
}
}
}
Which three are valid replacements for foo so that the program will compiled and run?

A. int i: array
B. int i = 0; i < 1; i++
C. ;;
D. ; i < 1; i++
E. ; i < 1;

Answer: ABC

NEW QUESTION 78
Given:



```
public class ScopeTest {
    int z;
    public static void main(String[] args) {
        ScopeTest myScope = new ScopeTest();
        int z = 6;
        System.out.println(z);
        myScope.doStuff();
        System.out.println(z);
        System.out.println(myScope.z);
    }
    void doStuff() {
        int z = 5;
        doStuff2();
        System.out.println(z);
    }
    void doStuff2() {
        z = 4;
    }
}
```

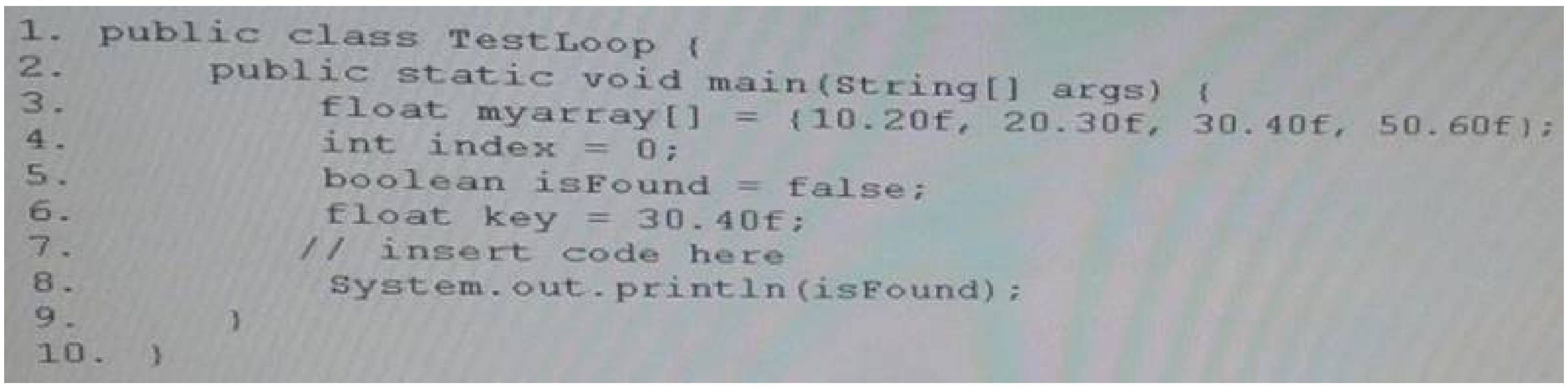
What is the result?

A. 6564
B. 6554
C. 6566
D. 6565

Answer: A

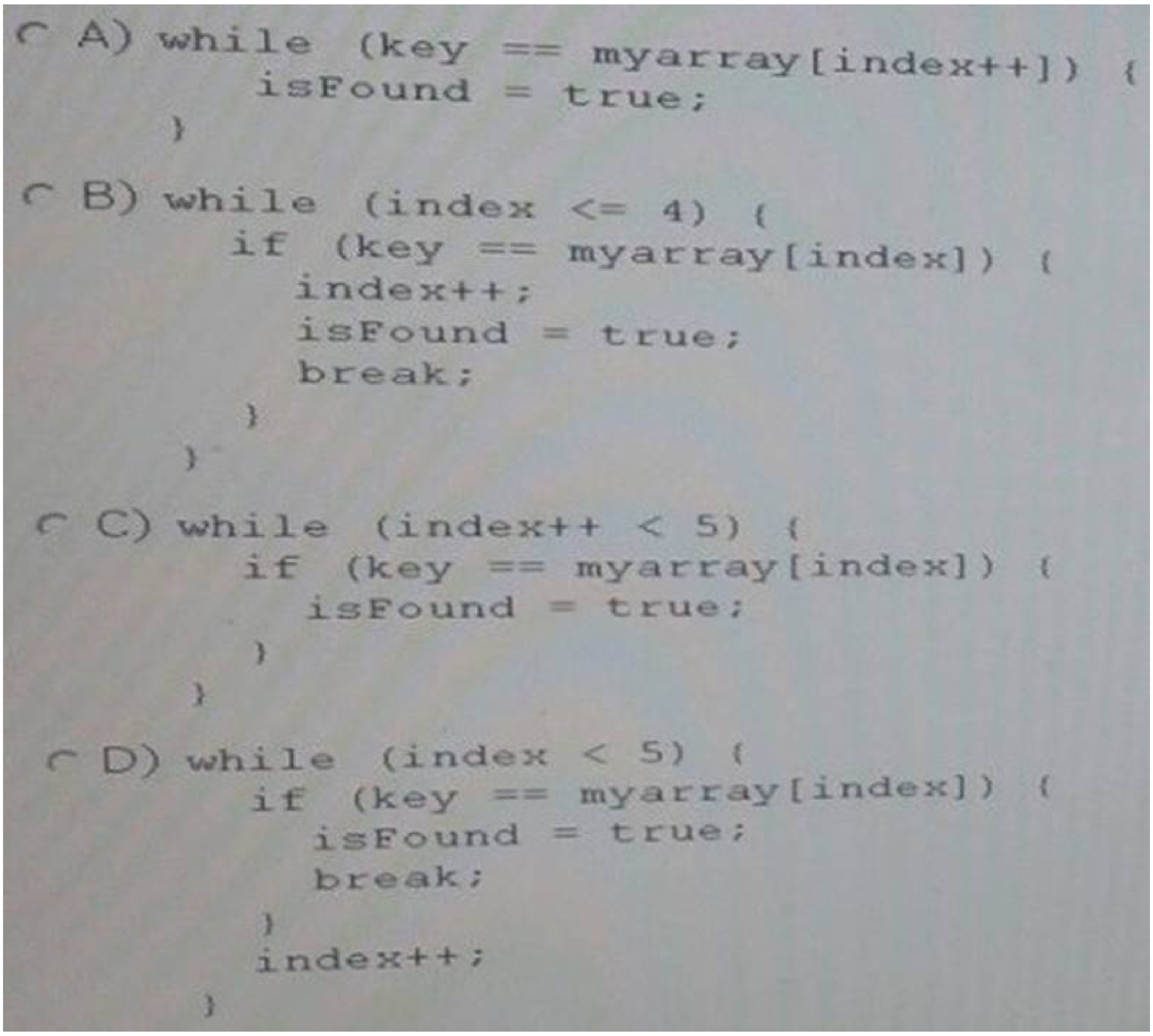
Explanation: Within main z is assigned 6. z is printed. Output: 6
Within doStuff z is assigned 5.DoStuff2 locally sets z to 4 (but MyScope.z is set to 4), but in Dostuff z is still 5. z is printed. Output: 5
Again z is printed within main (with local z set to 6). Output: 6
Finally MyScope.z is printed. MyScope.z has been set to 4 within doStuff2(). Output: 4

NEW QUESTION 79
Given:



```
1.  public class TestLoop {
2.      public static void main(String[] args) {
3.          float myarray[] = {10.20f, 20.30f, 30.40f, 50.60f};
4.          int index = 0;
5.          boolean isFound = false;
6.          float key = 30.40f;
7.          // insert code here
8.          System.out.println(isFound);
9.      }
10. }
```

Which code fragment, when inserted at line 7, enables the code print true?



```
○ A) while (key == myarray[index++]) {
         isFound = true;
     }

○ B) while (index <= 4) {
         if (key == myarray[index]) {
            index++;
            isFound = true;
            break;
         }
     }

○ C) while (index++ < 5) {
         if (key == myarray[index]) {
            isFound = true;
         }
     }

○ D) while (index < 5) {
         if (key == myarray[index]) {
            isFound = true;
            break;
         }
         index++;
     }
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: A

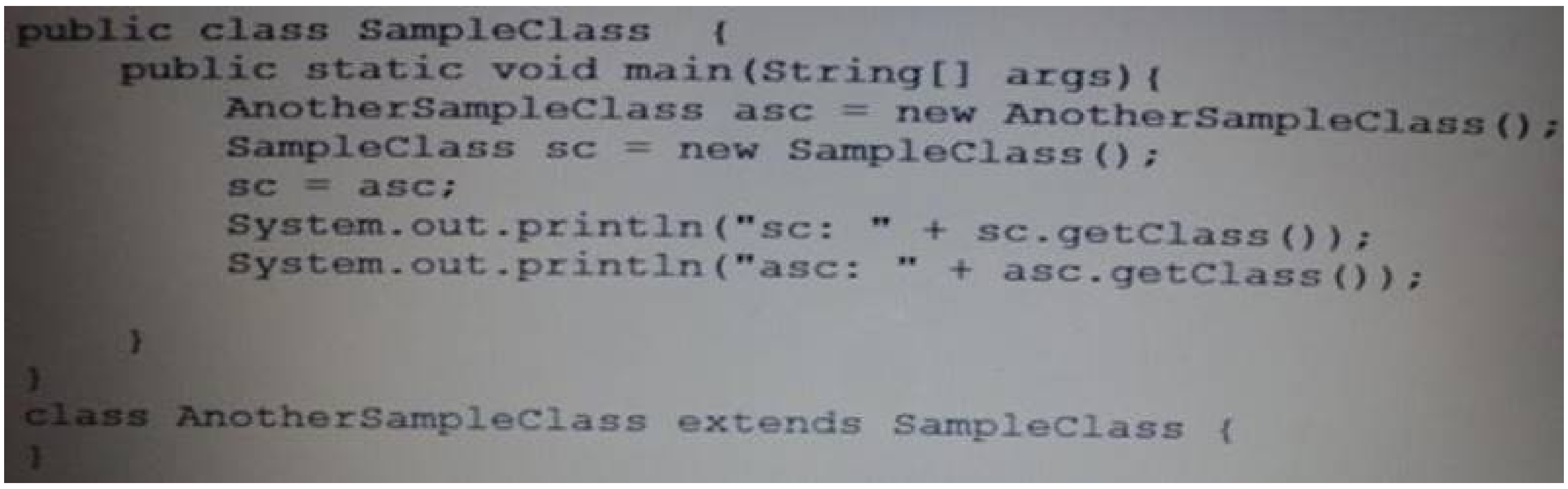
NEW QUESTION 80
Given the for loop construct:
for (expr1 ; expr2 ; expr3) { statement;
}
Which two statements aretrue?

A. This is not the only valid for loop construct; there exits another form of for loop constructor.
B. The expression expr1 is optiona
C. it initializes the loop and is evaluated once, as the loop begin.
D. When expr2 evaluates to false, the loop terminate
E. It is evaluated only after each iteration through the loop.
F. The expression expr3 must be presen
G. It is evaluated after each iteration through the loop.

Answer: BC

Explanation: The for statement have this forms: for (init-stmt;condition;next-stmt) { body
}
There are three clauses in the for statement.
The init-stmt statement is done before the loop is started, usually to initialize an iteration variable.
The condition expression is tested before each time the loop is done. The loop isn't executed if the boolean expression is false (the same as the while loop).
The next-stmt statement is done after the body is executed. It typically increments an iteration variable.

NEW QUESTION 82
Given:



```
public class SampleClass  {
     public static void main(String[] args){
          AnotherSampleClass asc = new AnotherSampleClass();
          SampleClass sc = new SampleClass();
          sc = asc;
          System.out.println("sc: " + sc.getClass());
          System.out.println("asc: " + asc.getClass());

     }
}
class AnotherSampleClass extends SampleClass {
}
```

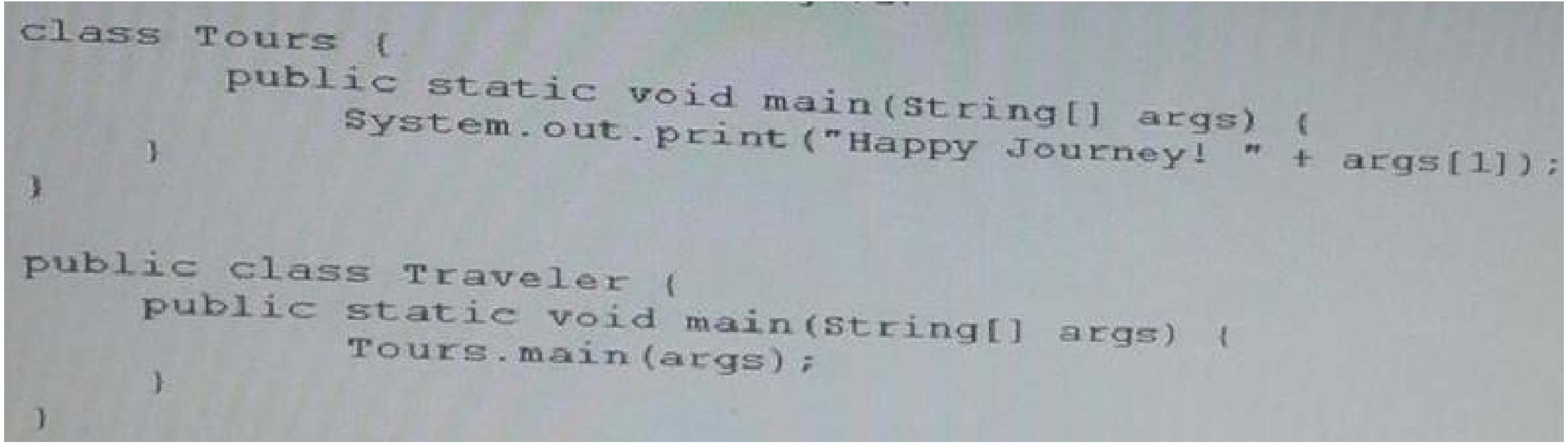
What is the result?

A. sc:class.Objectasc: class.AnotherSampleClass
B. sc: class.SampleClassasc: class.AnotherSampleClass
C. sc: class.AnotherSampleClass asc: class.SampleClass
D. sc: class.AnotherSampleClass asc: class.AnotherSampleClass

Answer: D

Explanation: Note: The getClass methodReturns the runtime class of an object. ThatClassobject is the object that is locked bystatic synchronizedmethods of the represented class.
Note: Because Java handles objects and arrays by reference, classes and array types are known as reference types.

NEW QUESTION 85
Given the code in a file Traveler.java:



```
class Tours {
          public static void main(String[] args) {
               System.out.print("Happy Journey! " + args[1]);
     }
}

public class Traveler {
     public static void main(String[] args) {
               Tours.main(args);
     }
}
```

And the commands:
Javac Traveler.java
Java Traveler Java Duke What is the result?

A. Happy Journey! Duke
B. Happy Journey! Java
C. An exception is thrown at runtime
D. The program fails to execute due to a runtime error

Answer: D

NEW QUESTION 86
Given:
public class MyFor {
public static void main(String[] args) {
for (int ii = 0; ii < 4; ii++) { System.out.println("ii = "+ ii); ii = ii +1;
}
}
}
What is the result?

A. ii = 0 ii = 2
B. ii =0ii = 1ii = 2ii = 3
C. ii =
D. Compilation fails.

Answer: A

NEW QUESTION 90
Which is a valid abstract class?

A. public abstract class Car { protected void accelerate();}
B. public interface Car {protected abstract void accelerate();}
C. public abstract class Car { protected final void accelerate();}
D. public abstract class Car { protected abstract void accelerate();}

E. public abstract class Car { protected abstract void accelerate() {//more car can do}}

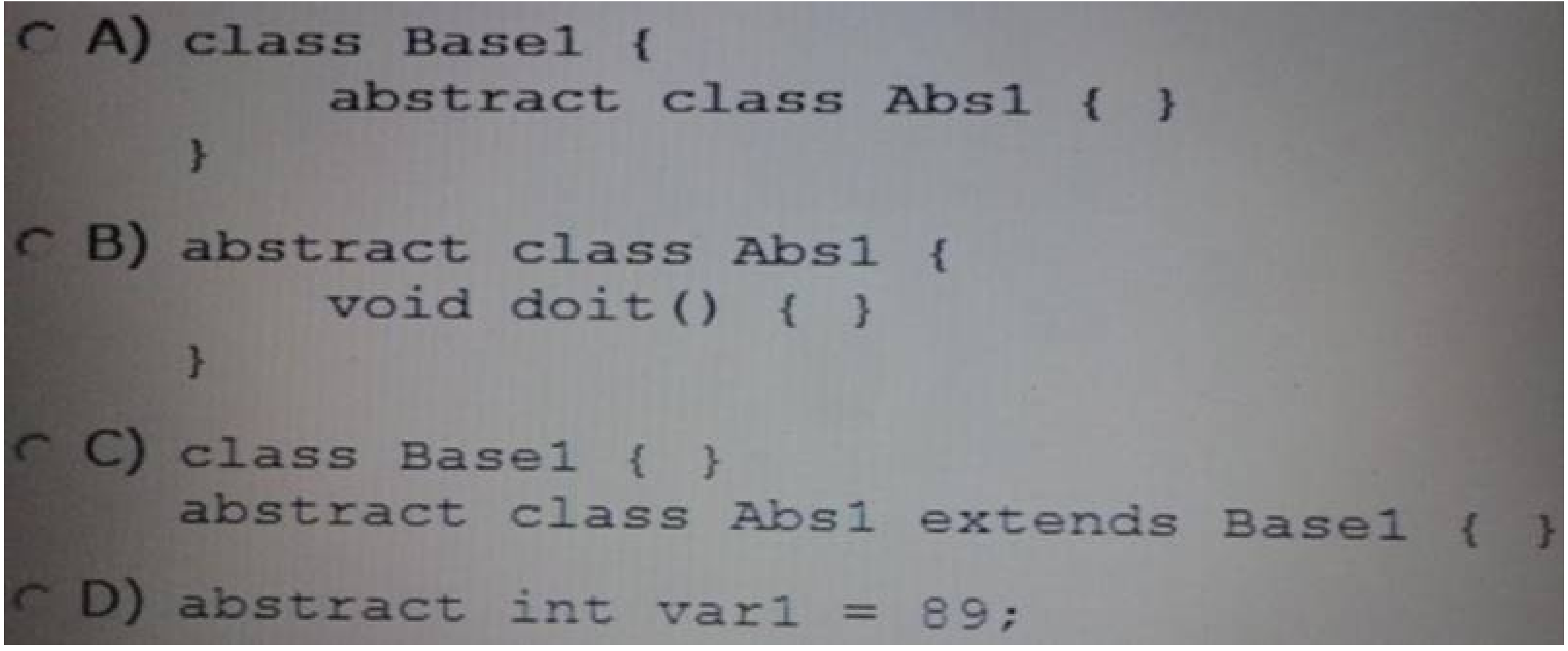
Answer: D

NEW QUESTION 93
boolean log3 = (5.0 != 6.0) && (4 != 5);
boolean log4 = (4 != 4)|| (4 == 4); System.out.println("log3:"+ log3 + \nlog4" + log4);
What is the result?

A. log3:false log4:true
B. log3:true log4:true
C. log3:true log4:false
D. log3:false log4:false

Answer: B

NEW QUESTION 97
Which code fragment is illegal?



```
○ A) class Base1 {
         abstract class Abs1 { }
     }

○ B) abstract class Abs1 {
         void doit() { }
     }

○ C) class Base1 { }
     abstract class Abs1 extends Base1 { }

○ D) abstract int var1 = 89;
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: D

Explanation: The abstract keyword cannot be used to declare an int variable.
The abstract keyword is used to declare a class or method to beabstract[3]. An abstract method has no implementation; all classes containing abstract methods must themselves be abstract, although not all abstract classes have abstract methods.

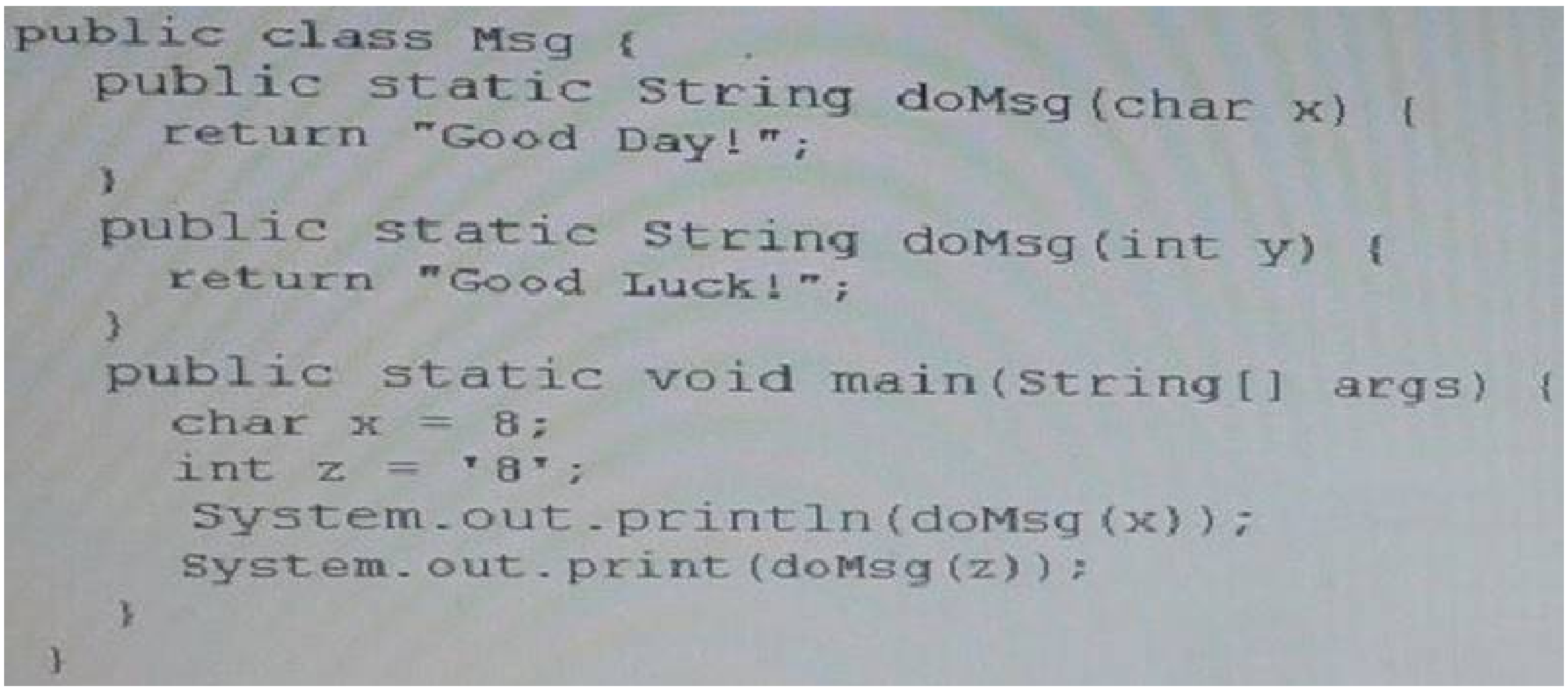
NEW QUESTION 101
The protected modifier on a Field declaration within a public class means that the field _______.

A. Cannot be modified
B. Can be read but not written from outside the class
C. Can be read and written from this class and its subclasses only within the same package
D. Can be read and written from this class and its subclasses defined in any package

Answer: D

Explanation: Reference:
http://beginnersbook.com/2013/05/java-access-modifiers/

NEW QUESTION 104
Given:



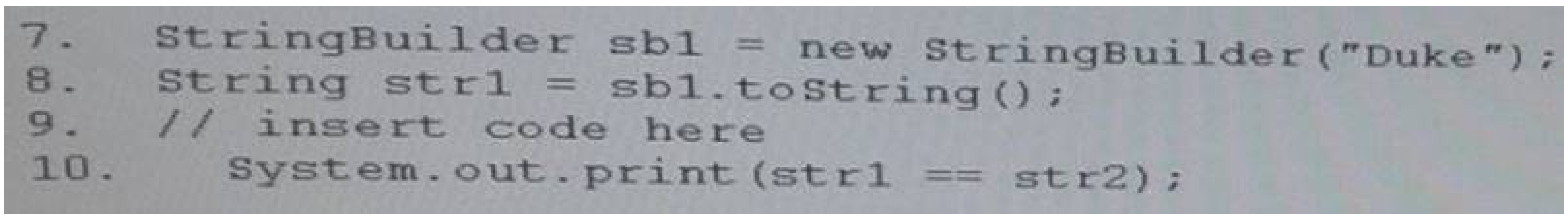
```
public class Msg {
  public static String doMsg(char x) {
    return "Good Day!";
  }
  public static String doMsg(int y) {
    return "Good Luck!";
  }
  public static void main(String[] args) {
    char x = 8;
    int z = '8';
    System.out.println(doMsg(x));
    System.out.print(doMsg(z));
  }
}
```

What is the result?

A. Good Day! Good Luck!
B. Good Day! Good Day!
C. Good Luck! Good Day!
D. Good Luck! Good Luck!
E. Compilation fails

Answer: E

NEW QUESTION 108
Given:



```
7.   StringBuilder sb1 = new StringBuilder("Duke");
8.   String str1 = sb1.toString();
9.   // insert code here
10.    System.out.print(str1 == str2);
```

Which code fragment, when inserted at line 9, enables the code to print true?

A. String str2 =str1;
B. String str2 = new string (str1);
C. String str2 = sb1.toString();
D. String str2 = “Duke”;

Answer: B

NEW QUESTION 113
Given:
class Overloading { int x(double d) {
System.out.println("one"); return 0;
}
String x(double d) { System.out.println("two"); return null;
}
double x(double d) { System.out.println("three"); return 0.0;
}
public static void main(String[] args) { new Overloading().x(4.0);
}
}
What is the result?

A. One
B. Two
C. Three
D. Compilation fails.

Answer: D

NEW QUESTION 118
Given:
class Sports { int num_players;
String name, ground_condition;
Sports(int np, String sname, String sground){ num_players = np;
name = sname; ground_condition = sground;
}
}
class Cricket extends Sports { int num_umpires;

int num_substitutes;
Which code fragment can be inserted at line //insert code here to enable the code to compile?

A. Cricket() {super(11, "Cricket", "Condidtion OK"); num_umpires =3; num_substitutes=2;}
B. Cricket() {super.ground_condition = "Condition OK"; super.name="Cricket"; super.num_players = 11;num_umpires =3; num_substitutes=2;}
C. Cricket() {this(3,2);super(11, "Cricket", "Condidtion OK");}Cricket(int nu, ns) { this.num_umpires =nu; this.num_substitutes=ns;}
D. Cricket() { this.num_umpires =3; this.num_substitutes=2;super(11, "Cricket", "Condidtion OK");}

Answer: A

Explanation: Incorrect:
not C, not D: call to super must be the first statement in constructor.

NEW QUESTION 122
A method is declared to take three arguments. A program calls this method and passes only two arguments. What is the result?

A. Compilation fails.
B. The third argument is given the value null.
C. The third argument is given the value void.
D. The third argument is given the value zero.
E. The third argument is given the appropriate false valuefor its declared type.
F. An exception occurs when the method attempts to access the third argument.

Answer: A

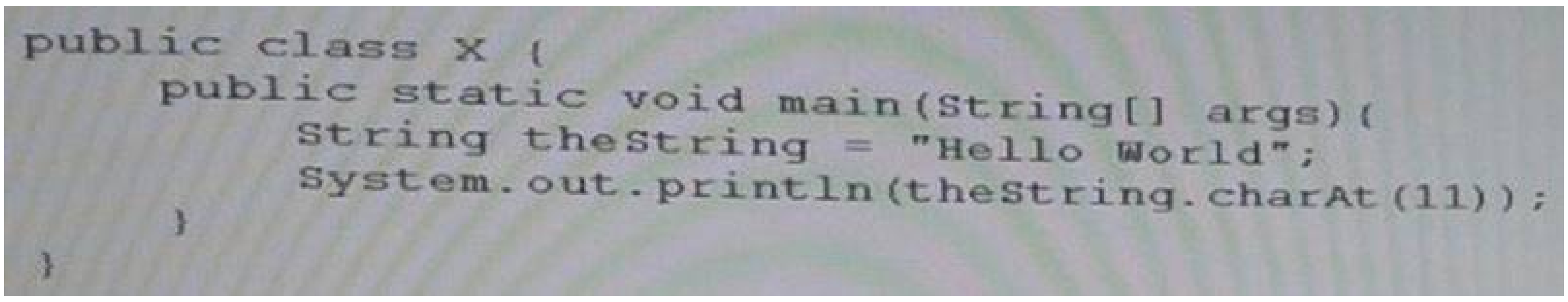
Explanation: The problem is noticed at build/compile time. At build you would receive an error message like:
required: int,int,int found: int,int

NEW QUESTION 123
Given:
public class X { static int i;
int j;
public static void main(String[] args) { X x1 = new X();
X x2 = new X(); x1.i = 3;
x1.j = 4;
x2.i = 5;
x2.j = 6;
System.out.println(x1.i + " "+
x1.j + " "+
x2.i + " "+
x2.j);
}
}
What is the result?

A. 3 4 5 6
B. 3 4 3 6
C. 5 4 5 6
D. 3 6 4 6

Answer: C

NEW QUESTION 126
Given:



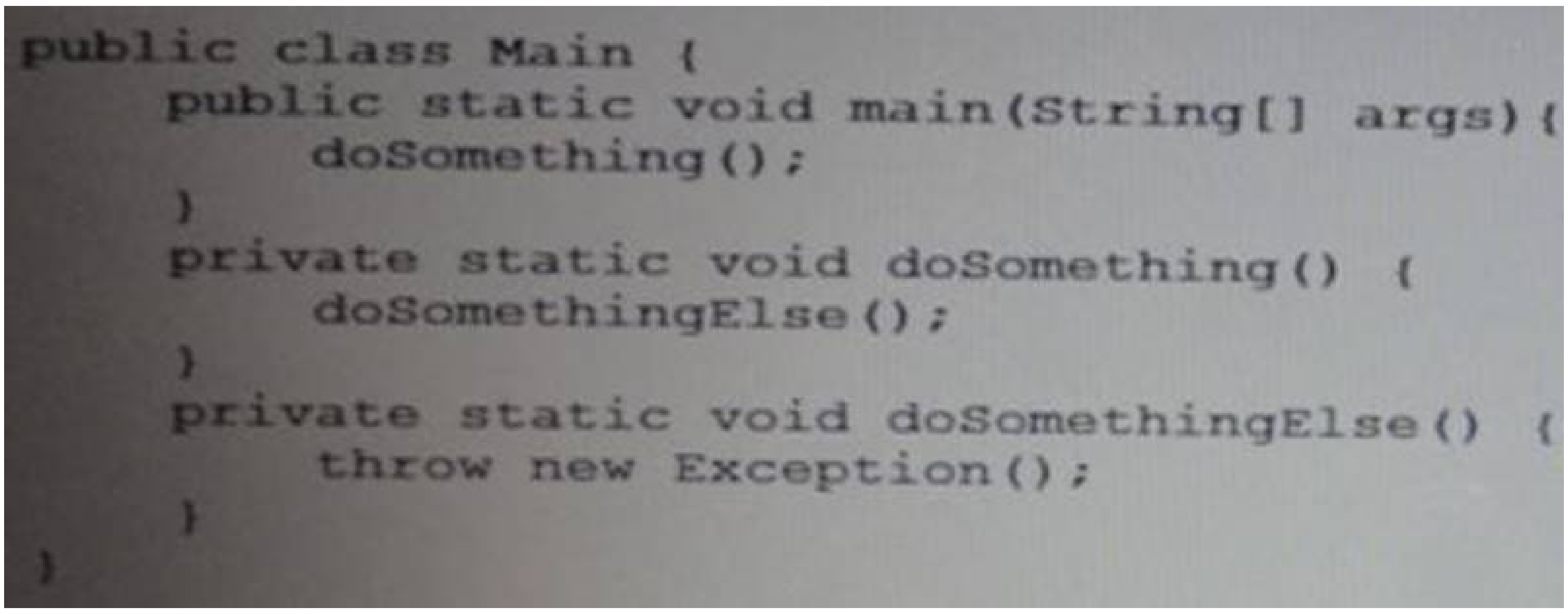
```
public class X {
     public static void main(String[] args){
          String theString = "Hello World";
          System.out.println(theString.charAt(11));
     }
}
```

What is the result?

A. The program prints nothing
B. d
C. A StringIndexOutOfBoundsException is thrown at runtime.
D. AnArrayIndexOutOfBoundsException is thrown at runtime.
E. A NullPointerException is thrown at runtime.

Answer: C

NEW QUESTION 129
Given:



```
public class Main {
    public static void main(String[] args){
        doSomething();
    }
    private static void doSomething() {
        doSomethingElse();
    }
    private static void doSomethingElse() {
        throw new Exception();
    }
}
```

Which approach ensures that the class can be compiled and run?

A. Put the throw new Exception() statement in the try block of try – catch
B. Put the doSomethingElse() method in the try block of a try – catch
C. Put the doSomething() method in the try block of a try – catch
D. Put thedoSomething() method and the doSomethingElse() method in the try block of a try – catch

Answer: A

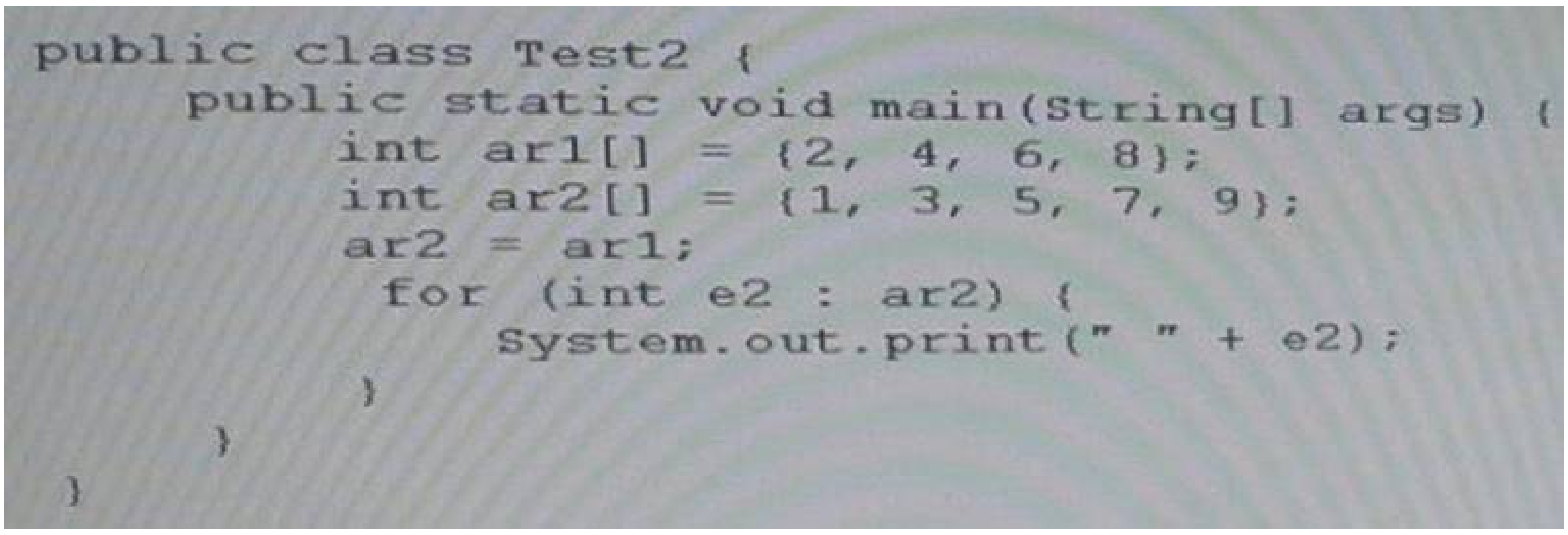
Explanation: We need to catch the exception in the doSomethingElse() method. Such as:
private static void doSomeThingElse() { try {
throw new Exception();} catch (Exception e)
{}
}
Note: One alternative, but not an option here, is the declare the exception in doSomeThingElse and catch it in the doSomeThing method.

NEW QUESTION 130
int [] array = {1,2,3,4,5}; for (int i: array) {
if (i < 2) { keyword1;
}
System.out.println(i); if (i == 3) { keyword2 ;
}}
What should keyword1 and keyword2 be respectively, in oreder to produce output 2345?

A. continue, break
B. break, break
C. break, continue
D. continue, continue

Answer: D

NEW QUESTION 131
Given:



```
public class Test2 {
    public static void main(String[] args) {
        int ar1[] = {2, 4, 6, 8};
        int ar2[] = {1, 3, 5, 7, 9};
        ar2 = ar1;
         for (int e2 : ar2) {
            System.out.print(" " + e2);
        }
    }
}
```

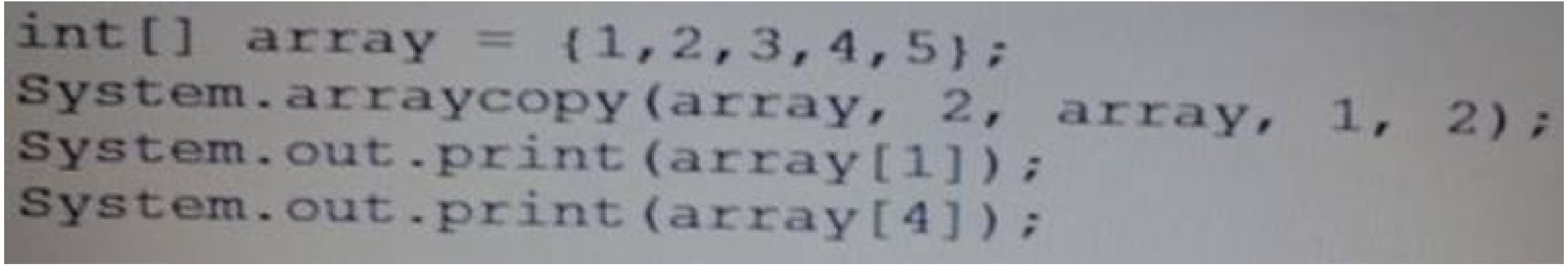
What is the result?

A. 2 4 6 8
B. 2 4 6 8 9
C. 1 3 5 7
D. 1 3 5 7 9

Answer: A

NEW QUESTION 132

Given the fragment:



```
int[] array = {1,2,3,4,5};
System.arraycopy(array, 2, array, 1, 2);
System.out.print(array[1]);
System.out.print(array[4]);
```

What is the result?

A. 14
B. 15
C. 24
D. 25
E. 34
F. 35

Answer: F

Explanation: The two elements 3 and 4 (starting from position with index 2) are copied into position index 1 and 2 in the same array.
Afterthe arraycopy command the array looks like:
{1, 3, 4, 4, 5};
Then element with index 1 is printed: 3 Then element with index 4 is printed: 5
Note:The System class has an arraycopy method that you can use to efficiently copy data from one array into another:
public static void arraycopy(Object src, int srcPos, Object dest, int destPos, int length)
The two Object arguments specify the array to copyfromand the array to copyto. The three int arguments specify the starting position in the source array, the starting position in the destination array, and the number of array elements to copy.

NEW QUESTION 133
A method is declared to take three arguments. A program calls this method and passes only two arguments. What is the results?

A. Compilation fails.
B. The third argument is given the value null.
C. The third argument is given the value void.
D. The third argument is given the value zero.
E. The third argument is given the appropriate falsy value for its declared typ
F. F) An exception occurs when the method attempts to access the third argument.

Answer: A

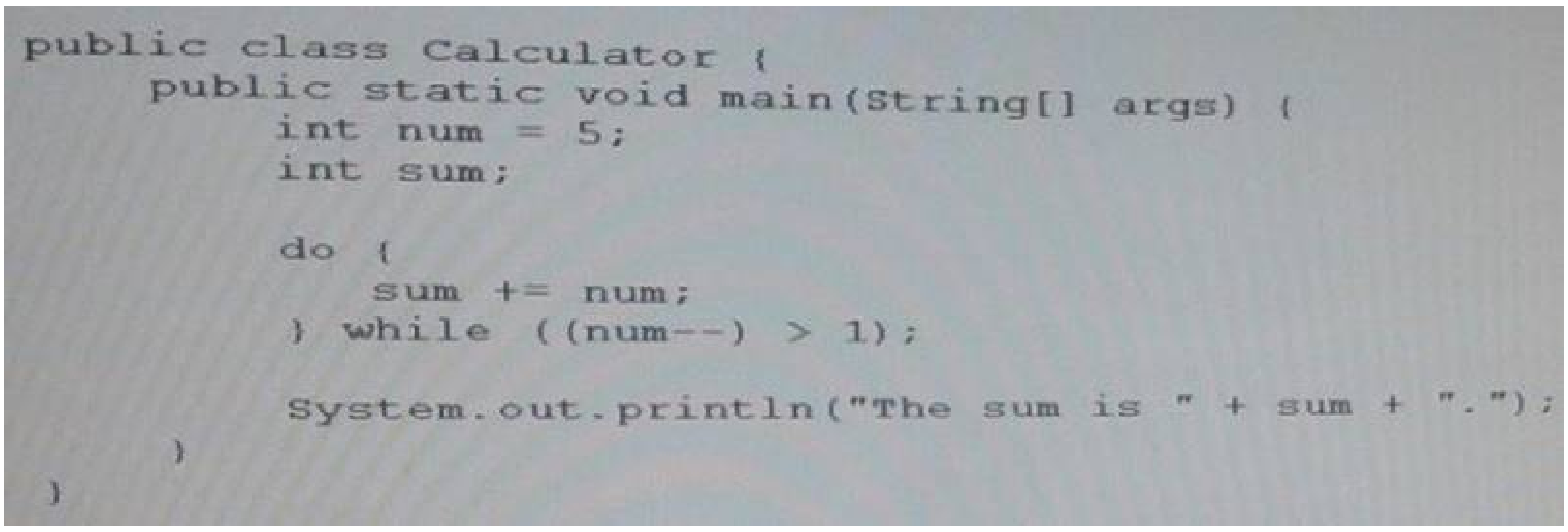
NEW QUESTION 138
Which two statements are true?

A. An abstract class can implement an interface.
B. An abstract class can be extended by an interface.
C. An interface CANNOT be extended by another interface.
D. An interface can be extended by an abstract class.
E. An abstract class can be extended by a concrete class.
F. An abstract class CANNOT be extended by an abstract class.

Answer: AE

Explanation: http://docs.oracle.com/javase/tutorial/java/IandI/abstract.html

NEW QUESTION 141
Given:



```
public class Calculator {
    public static void main(String[] args) {
        int num = 5;
        int sum;

        do {
            sum += num;
        } while ((num--) > 1);

        System.out.println("The sum is " + sum + ".");
    }
}
```

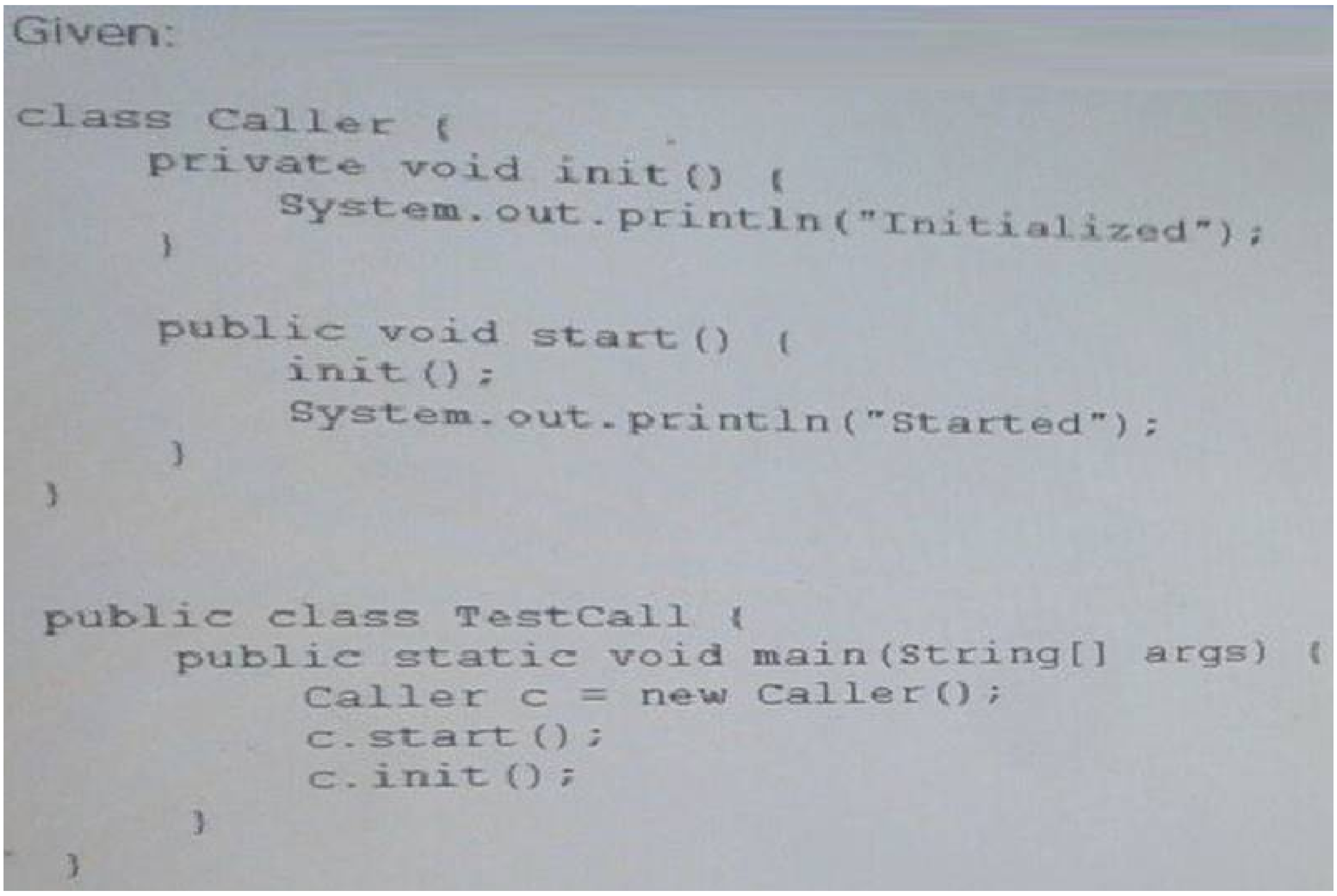
What is the result?

A. The sum is 2
B. The sum is 14

C. The sum is 15
D. The loop executes infinite times
E. Compilation fails

Answer: E

NEW QUESTION 146
Given:



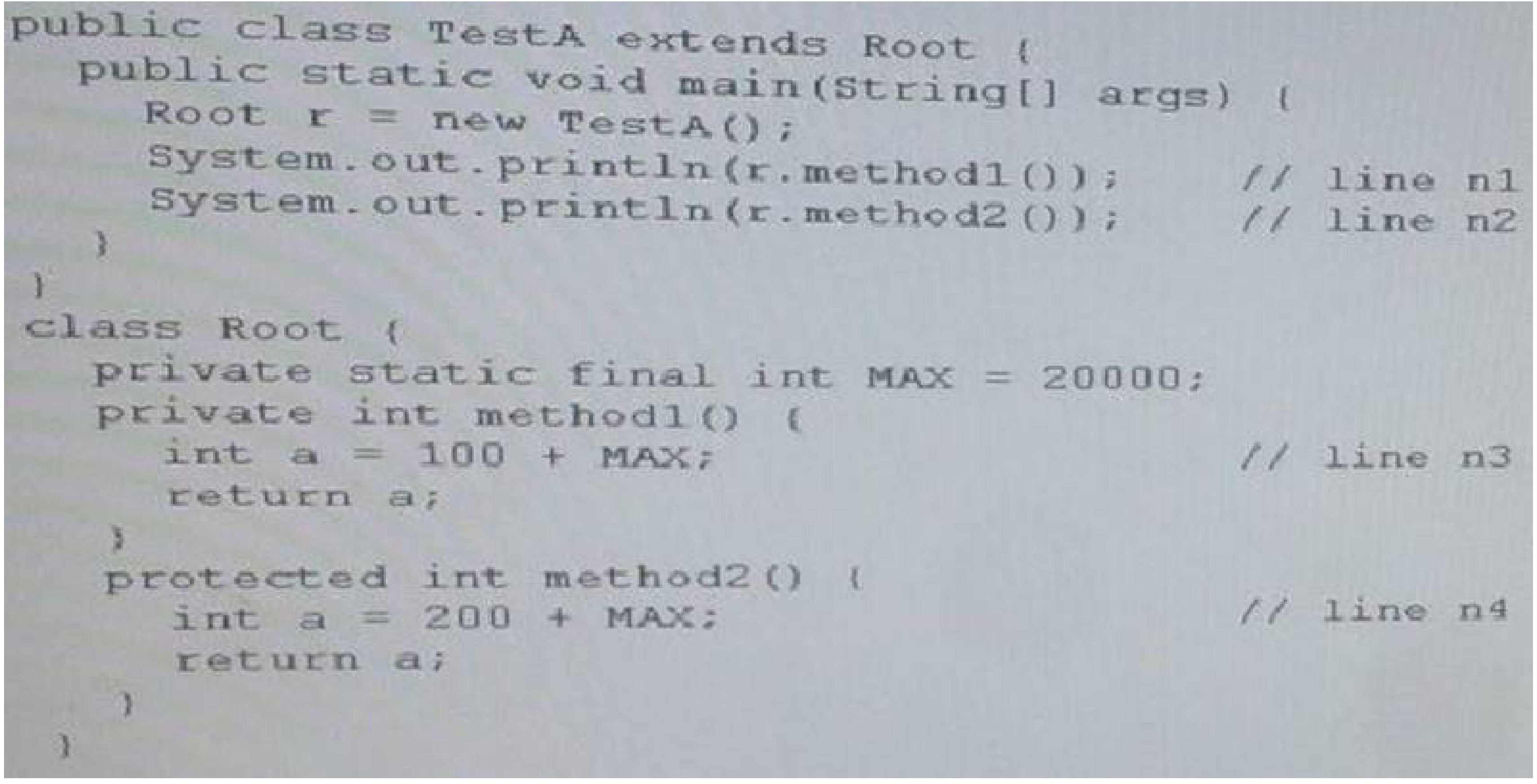
```
Given:

class Caller {
    private void init() {
        System.out.println("Initialized");
    }

    public void start() {
        init();
        System.out.println("Started");
    }
}

public class TestCall {
    public static void main(String[] args) {
        Caller c = new Caller();
        c.start();
        c.init();
    }
}
```

What is the result?

A. Initialized Started
B. Initialized Started Initialized
C. Compilation fails
D. An exception is thrown at runtime

Answer: B

NEW QUESTION 150
Given the fragments:



```
public class TestA extends Root {
  public static void main(String[] args) {
    Root r = new TestA();
    System.out.println(r.method1());     // line n1
    System.out.println(r.method2());     // line n2
  }
}
class Root {
  private static final int MAX = 20000;
  private int method1() {
    int a = 100 + MAX;                   // line n3
    return a;
  }
  protected int method2() {
    int a = 200 + MAX;                   // line n4
    return a;
  }
}
```

Which line causes a compilation error?

A. Line n1
B. Line n2

C. Line n3
D. Line n4

Answer: A

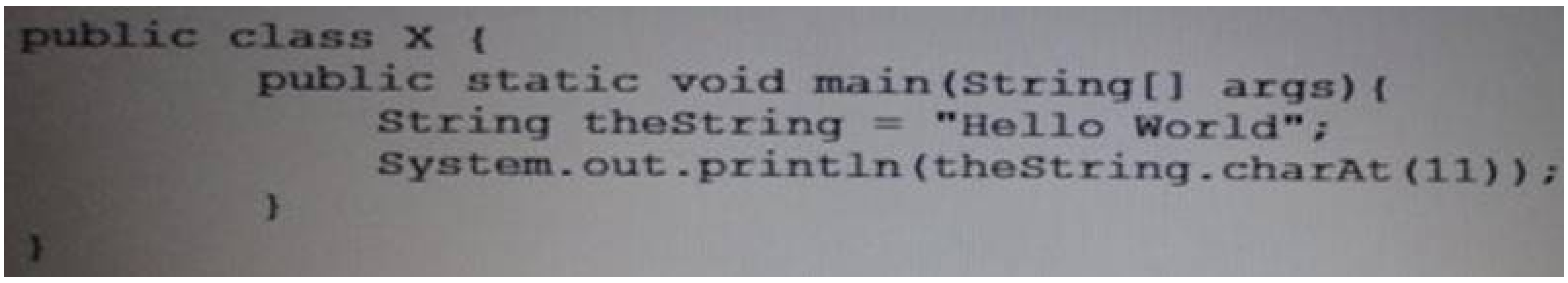
NEW QUESTION 152
Given:
public class ColorTest {
public static void main(String[] args) {
String[] colors = {"red", "blue","green","yellow","maroon","cyan"}; int count = 0;
for (String c : colors) { if (count >= 4) { break;
}
else { continue;
}
if (c.length() >= 4) { colors[count] = c.substring(0,3);
}
count++;
}
System.out.println(colors[count]);
}
}
What is the result?

A. Yellow
B. Maroon
C. Compilation fails
D. A StringIndexOutOfBoundsException is thrown at runtime.

Answer: C

Explanation: Theline, if (c.length() >= 4) {, is never reached. This causes a compilation error.
Note: The continue statement skips the current iteration of a for, while , or do-while loop. An unlabeled break statement terminates the innermost switch, for, while, or do- while statement, but a labeled break terminates an outer statement.

NEW QUESTION 154
Given:



```
public class X {
        public static void main(String[] args){
            String theString = "Hello World";
            System.out.println(theString.charAt(11));
        }
}
```

What is the result?

A. There is no output
B. d is output
C. A StringIndexOutOfBoundsException is thrown at runtime
D. An ArrayIndexOutOfBoundsException is thrown at runtime
E. A NullPointException is thrown at runtime
F. A StringArrayIndexOutOfBoundsException is thrown at runtime

Answer: C

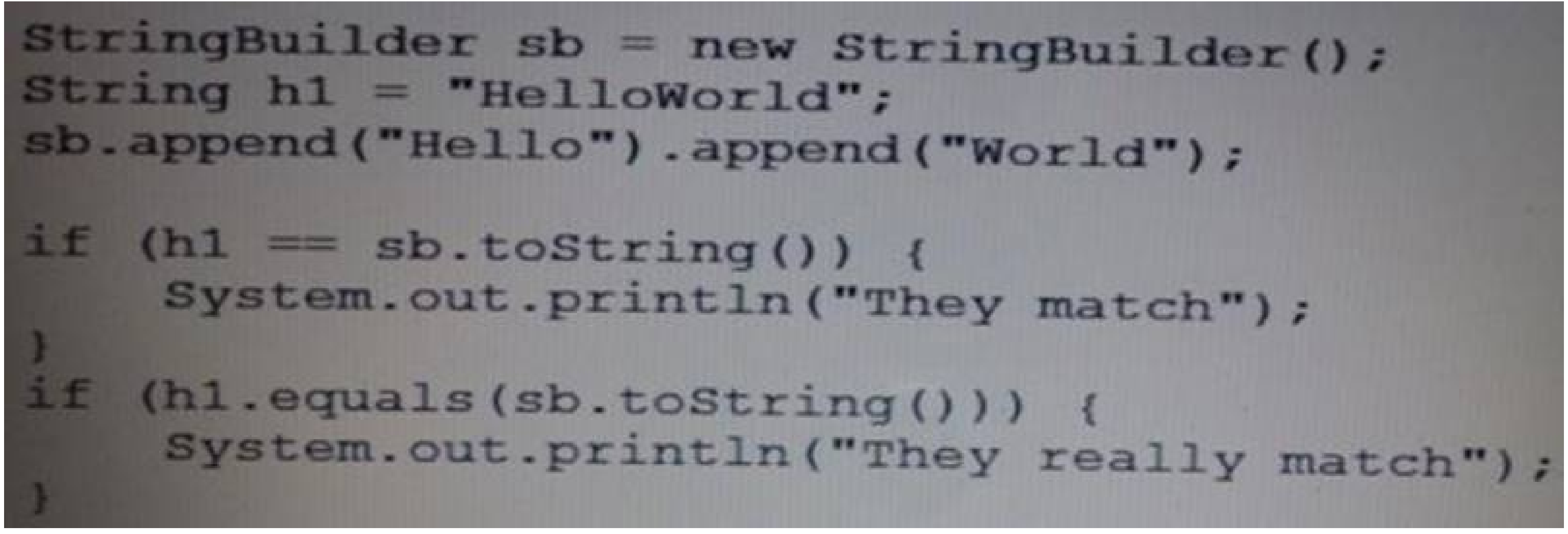
Explanation: There are only 11 characters in the string "Hello World". The code theString.charAt(11)
retrieves the 12th character, which does not exist. A StringIndexOutOfBoundsException is thrown.
Exception in thread "main" java.lang.StringIndexOutOfBoundsException: String index out of range: 11

NEW QUESTION 159
View the exhibit:
public class Student { public String name= ""; public int age = 0;
public String major = "Undeclared"; public boolean fulltime = true; public void display() {
System.out.println("Name: " + name + " Major: " + major); } public boolean isFullTime() {
return fulltime;
}
}
Which line of code initializes a student instance?

A. Student student1;
B. Student student1 = Student.new();
C. Student student1 = new Student();
D. Student student1 = Student();

Answer: C

NEW QUESTION 163
Given a code fragment:



```
StringBuilder sb = new StringBuilder();
String h1 = "HelloWorld";
sb.append("Hello").append("World");

if (h1 == sb.toString()) {
    System.out.println("They match");
}
if (h1.equals(sb.toString())) {
    System.out.println("They really match");
}
```

What is the result?

A. They match They real match
B. They really match
C. They match
D. Nothing is printed to the screen

Answer: B

NEW QUESTION 166
Given the code fragment:
public class ForTest {
public static void main(String[] args) { int[] array = {1, 2, 3};
for (foo) {
}
}
Which three code fragments,when replaced individually forfoo, enables the program to compile?

A. int i : array
B. int i = 0; i < 1;
C. ; ;
D. ; i < 1; i++
E. i = 0; i<1;

Answer: ABC

NEW QUESTION 168
Given:
Class A { } Class B { }
Interface X { } Interface Y { }
Which two definitions of class C are valid?

A. Class C extends A implements X { }
B. Class C implements Y extends B { }
C. Class C extends A, B { }
D. Class C implements X, Y extends B { }
E. Class C extends B implements X, Y { }

Answer: AE

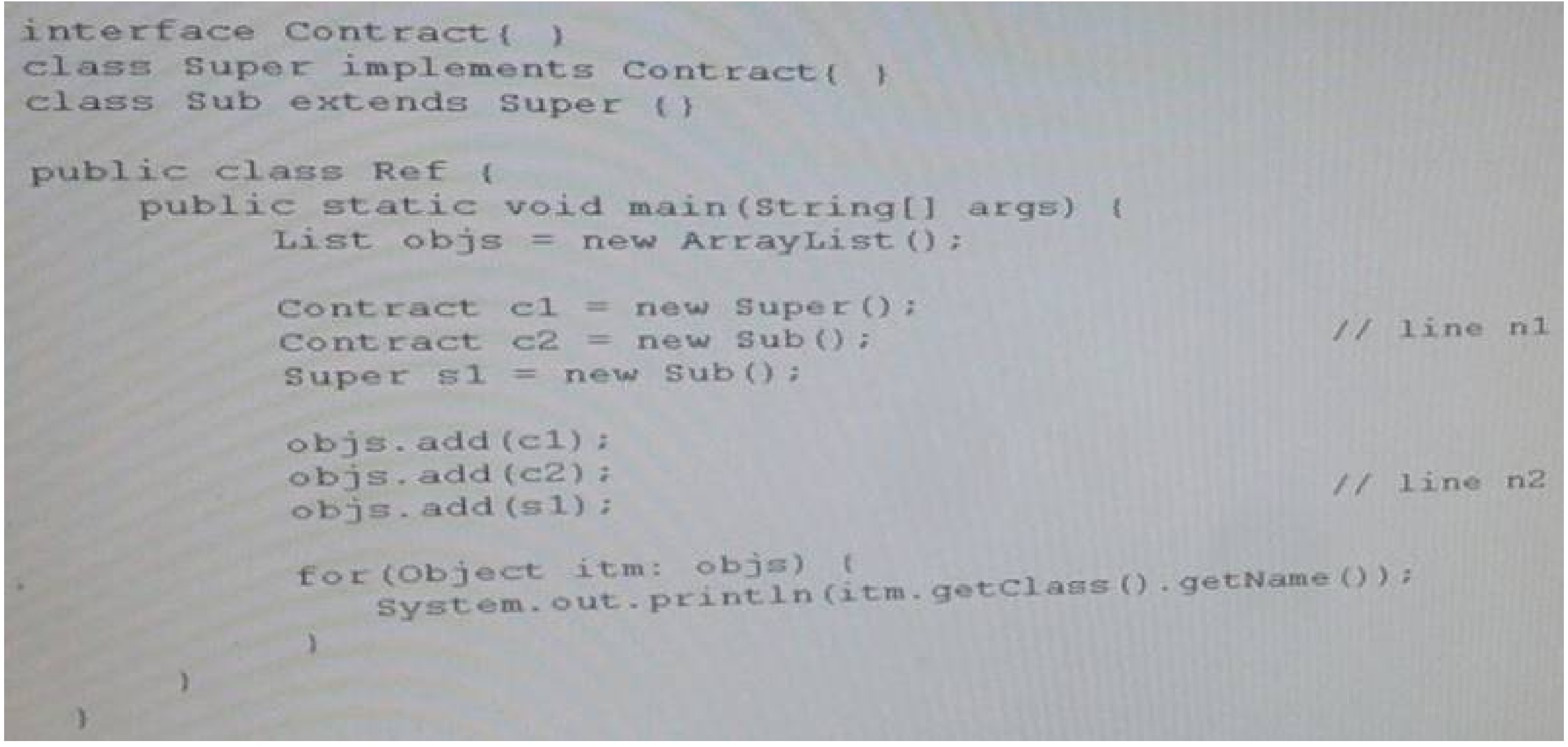
Explanation: extends is for extending a class.
implements is for implementing an interface.
Java allows for a class to implement many interfaces.

NEW QUESTION 173
Given:
public class TestOperator {
public static voidmain(String[] args) { int result = 30 -12 / (2*5)+1; System.out.print("Result = " + result);
}
}
What is the result?

A. Result = 2
B. Result = 3
C. Result = 28
D. Result = 29
E. Result = 30

Answer: E

NEW QUESTION 178
Given the code fragments:



```
interface Contract{ }
class Super implements Contract{ }
class Sub extends Super {}

public class Ref {
    public static void main(String[] args) {
        List objs = new ArrayList();

        Contract c1 = new Super();
        Contract c2 = new Sub();                     // line n1
        Super s1 = new Sub();

        objs.add(c1);
        objs.add(c2);                                // line n2
        objs.add(s1);

        for(Object itm: objs) {
            System.out.println(itm.getClass().getName());
        }
    }
}
```

What is the result?

A. Super SubSub
B. Contract Contract Super
C. Compilation fails at line n1
D. Compilation fails at line n2

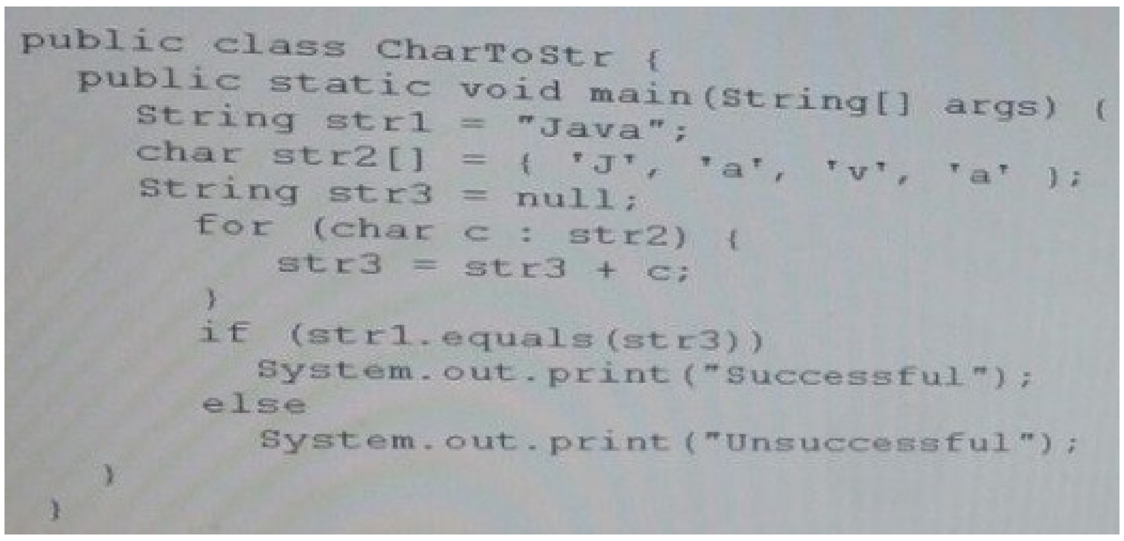
Answer: D

NEW QUESTION 182
Given the code fragment:
for (int ii = 0; ii < 3;ii++) { int count = 0;
for (int jj = 3; jj > 0; jj--) { if (ii == jj) {
++count; break;
}
}
System.out.print(count); continue;
}
What is the result?

A. 011
B. 012
C. 123
D. 000

Answer: A

NEW QUESTION 185
Given:



```
public class CharToStr {
  public static void main(String[] args) {
    String str1 = "Java";
    char str2[] = { 'J', 'a', 'v', 'a' };
    String str3 = null;
      for (char c : str2) {
        str3 = str3 + c;
      }
      if (str1.equals(str3))
        System.out.print("Successful");
      else
        System.out.print("Unsuccessful");
  }
}
```

What is result?

A. Successful
B. Unsuccessful
C. Compilation fails
D. An exception is thrown at runtime

Answer: C

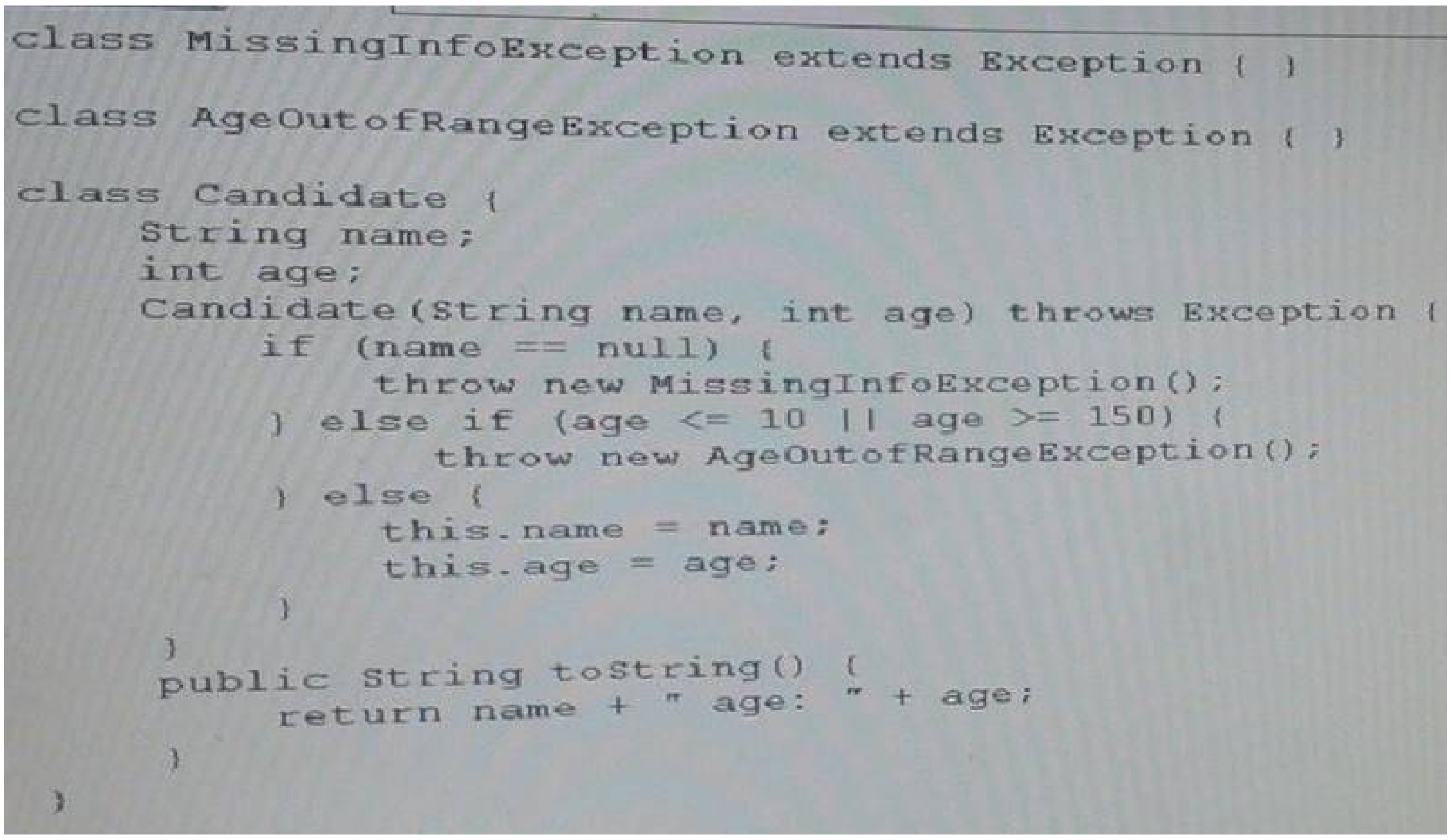
NEW QUESTION 187

Given:
package p1; public class Test {
static double dvalue; static Test ref;
public static void main(String[] args) { System.out.println(ref); System.out.println(dvalue);
}
}
What is the result?

A. p1.Test.class 0.0
B. <the summary address refrenced by ref> 0.000000
C. Null 0.0
D. Compilation fails
E. A NullPointerException is thrown at runtime

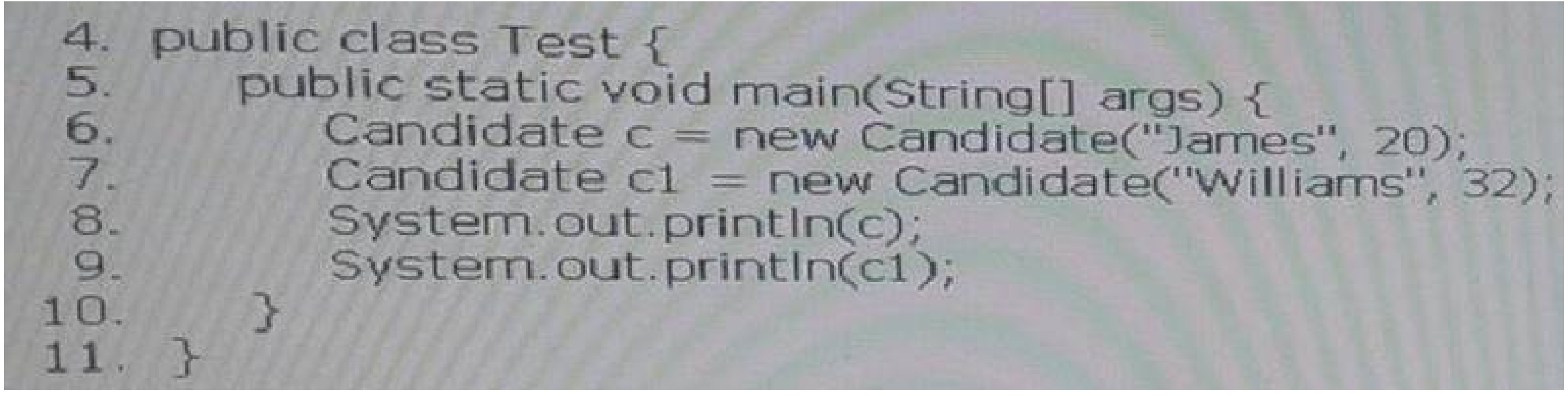
Answer: C

NEW QUESTION 188
View the exhibit.



```
class MissingInfoException extends Exception { }

class AgeOutofRangeException extends Exception { }

class Candidate {
    String name;
    int age;
    Candidate(String name, int age) throws Exception {
        if (name == null) {
            throw new MissingInfoException();
        } else if (age <= 10 || age >= 150) {
            throw new AgeOutofRangeException();
        } else {
            this.name = name;
            this.age = age;
        }
    }
    public String toString() {
        return name + " age: " + age;
    }
}
```

Given the code fragment:



```
4.  public class Test {
5.      public static void main(String[] args) {
6.          Candidate c = new Candidate("James", 20);
7.          Candidate c1 = new Candidate("Williams", 32);
8.          System.out.println(c);
9.          System.out.println(c1);
10.     }
11. }
```

Which change enables the code to print the following? James age: 20
Williams age: 32

A. Replacing line 5 with public static void main (String [] args) throws MissingInfoException, AgeOutofRangeException {
B. Replacing line 5 with public static void main (String [] args) throws.Exception {
C. Enclosing line 6 and line 7 within a try block and adding: catch(Exception e1) { //code goes here}catch (missingInfoExceptione2) { //code goes here} catch (AgeOutofRangeException e3) {//code goes here}
D. Enclosing line 6 and line 7 within a try block and adding: catch (missingInfoException e2) { //code goes here}catch (AgeOutofRangeException e3) {//code goes here}

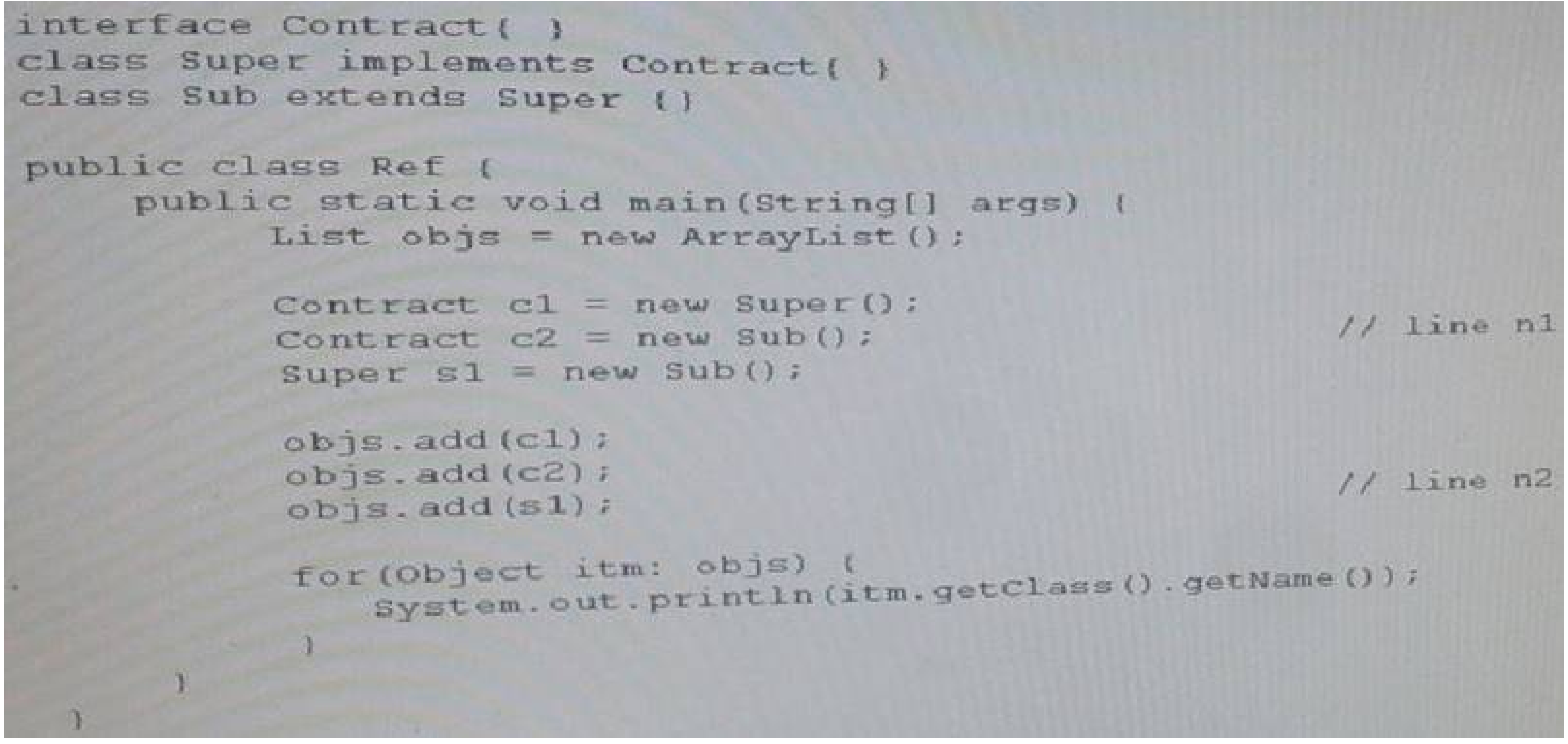
Answer: C

NEW QUESTION 189
Given the code fragment:
String name = "Spot"; int age = 4;
String str ="My dog " + name + " is " + age; System.out.println(str);
And
StringBuilder sb = new StringBuilder();
Using StringBuilder, which code fragment is the best potion to build and print the following
string My dog Spot is 4

A. sb.append("My dog " + name + " is " + age); System.out.println(sb);
B. sb.insert("My dog ").append(name + " is " + age); System.out.println(sb);
C. sb.insert("My dog ").insert(name).insert(" is ").insert(age); System.out.println(sb);
D. sb.append("My dog ").append(name).append(" is ").append(age); System.out.println(sb);

Answer: AD

NEW QUESTION 193
Given the code fragment:



```
interface Contract{ }
class Super implements Contract{ }
class Sub extends Super {}

public class Ref {
    public static void main(String[] args) {
        List objs = new ArrayList();

        Contract c1 = new Super();
        Contract c2 = new Sub();                  // line n1
        Super s1 = new Sub();

        objs.add(c1);
        objs.add(c2);                             // line n2
        objs.add(s1);

        for(Object itm: objs) {
            System.out.println(itm.getClass().getName());
        }
    }
}
```

A. Super SubSub
B. Contract Contract Super
C. Compilation fails at line n1
D. Compilation fails at line n2

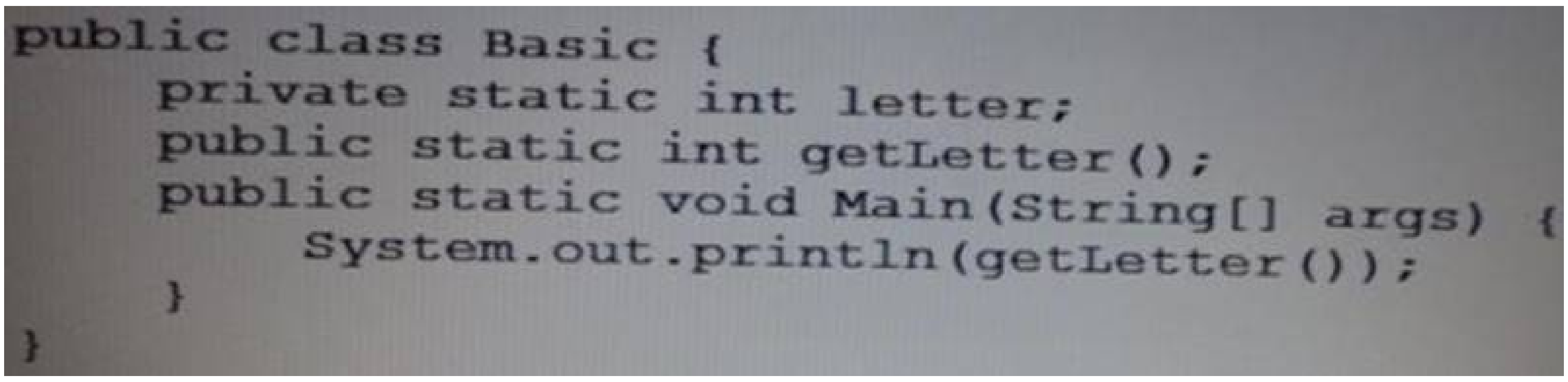
Answer: D

NEW QUESTION 197
Given:
public class FieldInit { char c;
booleanb; float f;
void printAll() { System.out.println("c = " + c); System.out.println("c = " + b); System.out.println("c = " + f);
}
public static void main(String[] args) { FieldInit f = new FieldInit(); f.printAll();
}
}
What is the result?

A. c = null b = falsef = 0.0F
B. c = 0b = false f = 0.0f
C. c = null b = truef = 0.0
D. c =b = false f = 0.0

Answer: D

NEW QUESTION 202
Given:



```
public class Basic {
    private static int letter;
    public static int getLetter();
    public static void Main(String[] args) {
        System.out.println(getLetter());
    }
}
```

Why will the code not compile?

A. A static field cannot be private.
B. The getLetter method has no body.
C. There is no setLetter method.
D. The letter field is uninitialized.
E. It contains a method named Main instead of ma

Answer: B

Explanation: The getLetter() method needs a body public static int getLetter() { }; .

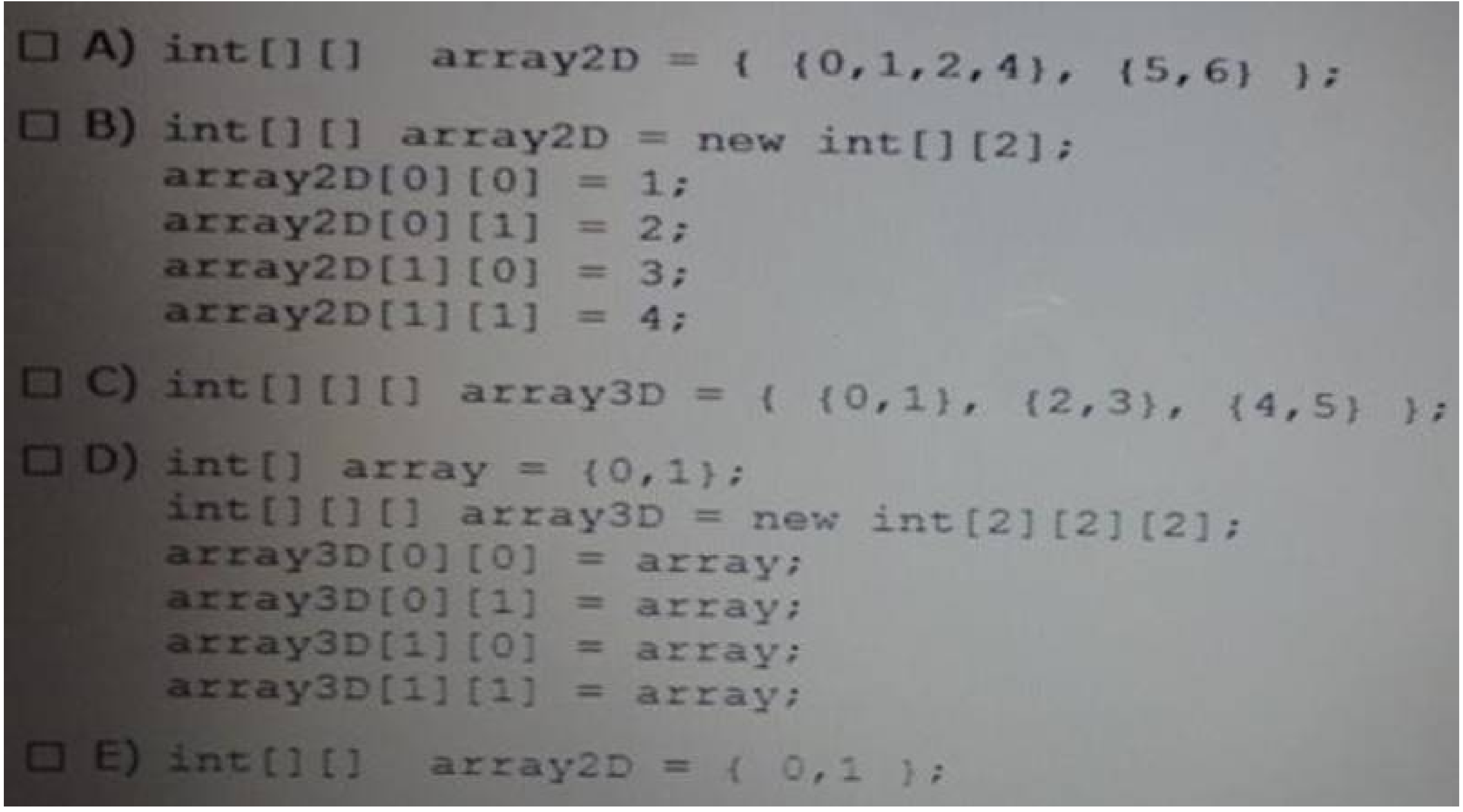
NEW QUESTION 205
Given:
public class Test {
public static void main(String[] args) { int arr[] = new int[4];
arr[0] = 1;
arr[1] = 2;
arr[2] = 4;
arr[3] = 5;
int sum = 0; try {
for (int pos = 0; pos <= 4; pos++) { sum = sum +arr[pos];
}
} catch (Exception e) { System.out.println("Invalid index");
}
System.out.println(sum);
}
}
What is the result?

A. 12
B. Invalid Index 12
C. Invalid Index
D. Compilation fails

Answer: B

Explanation: The loop (for (int pos = 0; pos <= 4; pos++) {), it should be pos <= 3, causes an exception, which is caught. Then the correct sum is printed.

NEW QUESTION 208
Which two are valid instantiations and initializations of a multi dimensional array?



```
□ A) int[][]  array2D = { {0,1,2,4}, {5,6} };
□ B) int[][] array2D = new int[][2];
         array2D[0][0] = 1;
         array2D[0][1] = 2;
         array2D[1][0] = 3;
         array2D[1][1] = 4;
□ C) int[][][] array3D = { {0,1}, {2,3}, {4,5} };
□ D) int[] array = {0,1};
         int[][][] array3D = new int[2][2][2];
         array3D[0][0] = array;
         array3D[0][1] = array;
         array3D[1][0] = array;
         array3D[1][1] = array;
□ E) int[][]  array2D = { 0,1 };
```

A. Option A
B. Option B
C. Option C
D. Option D
E. Option E

Answer: AD

NEW QUESTION 213
Given:
public class TestLoop {
public static void main(String[] args) { int array[] = {0, 1, 2, 3, 4};
int key = 3;
for (int pos = 0; pos < array.length; ++pos) { if (array[pos] == key) {
break;

```
}
}
System.out.print("Found " + key + "at " + pos);
}
}
```

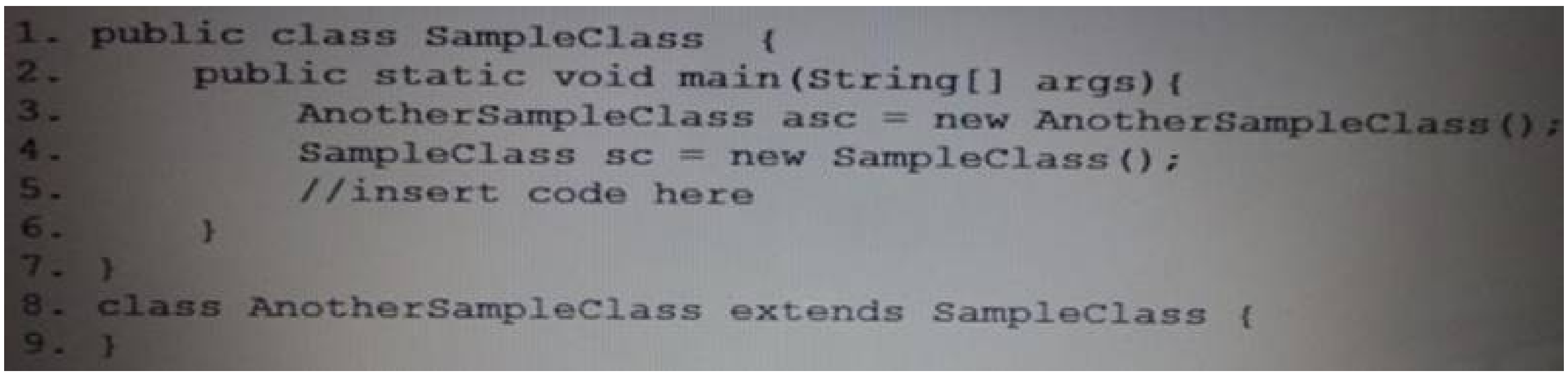
What is the result?

A. Found 3 at 2
B. Found 3 at 3
C. Compilation fails
D. An exception is thrown at runtime

Answer: C

Explanation: The following linedoes not compile: System.out.print("Found " + key + "at " + pos);
The variable pos is undefined at this line, as its scope is only valid in the for loop. Any variables created inside of a loop are LOCAL TO THE LOOP.

NEW QUESTION 218
Given:



```
1. public class SampleClass  {
2.     public static void main(String[] args){
3.         AnotherSampleClass asc = new AnotherSampleClass();
4.         SampleClass sc = new SampleClass();
5.         //insert code here
6.     }
7. }
8. class AnotherSampleClass extends SampleClass {
9. }
```

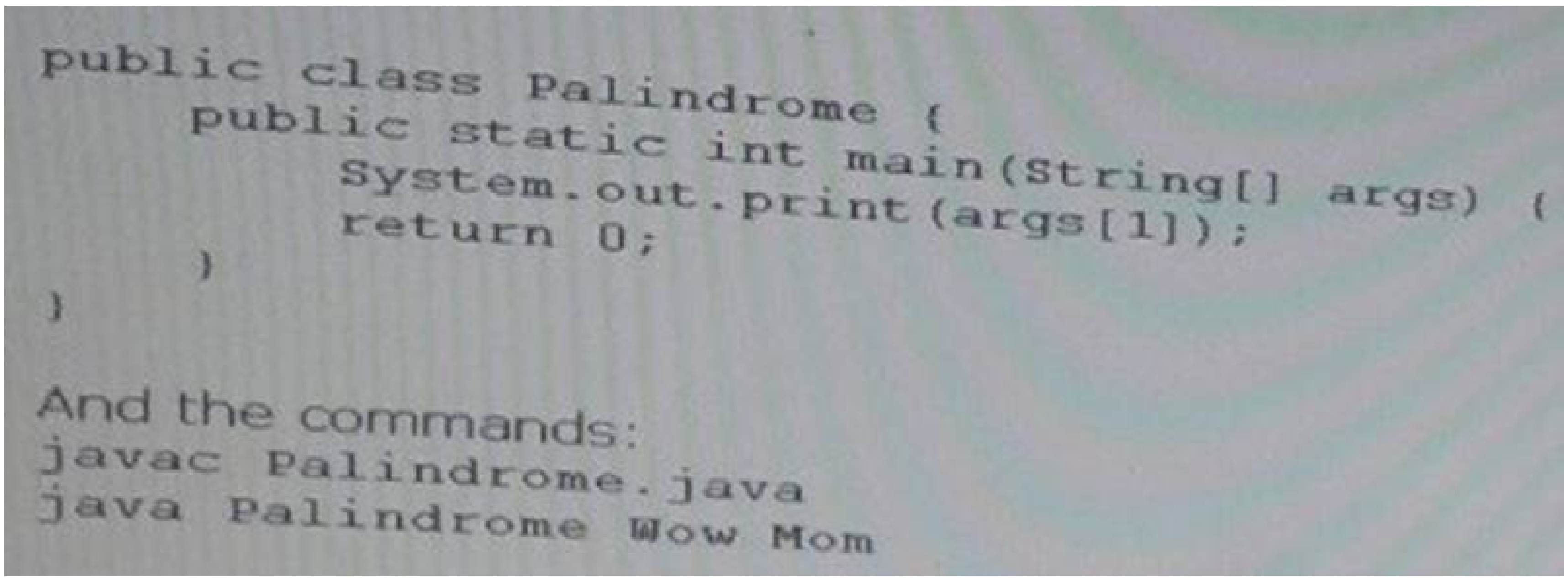
Which statement, when inserted into line 5, is valid change?

A. asc = sc;
B. sc = asc;
C. asc = (object) sc;
D. asc = sc.clone ()

Answer: B

Explanation: Works fine.

NEW QUESTION 219
Given:



```
public class Palindrome {
    public static int main(String[] args) {
        System.out.print(args[1]);
        return 0;
    }
}
```

And the commands:

```
javac Palindrome.java
java Palindrome Wow Mom
```

What is the result?

A. Compilation fails
B. The code compiles, but does not execute.
C. Paildrome
D. Wow
E. Mom

Answer: B

NEW QUESTION 224
Which two are valid declarations of a two-dimensional array?

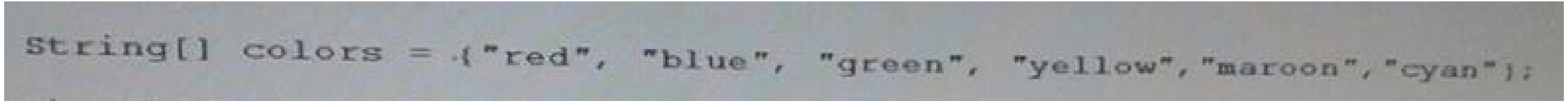
A. int [] [] array2D;
B. int [2] [2] array2D;
C. int array2D [];

D. int [] array2D [];
E. int [] [] array2D [];

Answer: AD

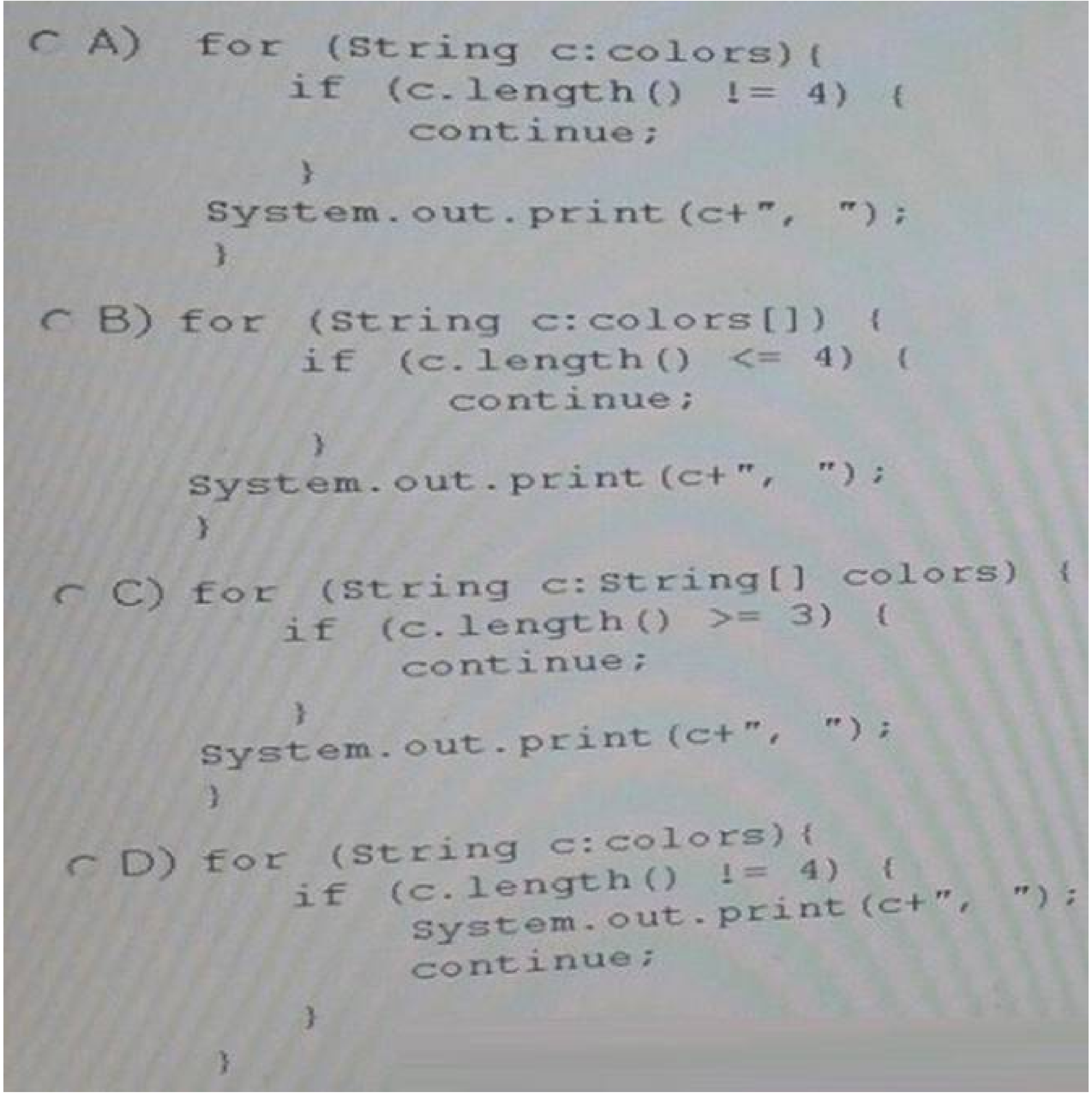
Explanation: int[][] array2D; is the standard convention to declare a 2-dimensional integer array. int[] array2D[]; works as well, but it is not recommended.

NEW QUESTION 229
Given the code fragment:



```
String[] colors = {"red", "blue", "green", "yellow","maroon","cyan"};
```

Which code fragment prints blue, cyan, ?

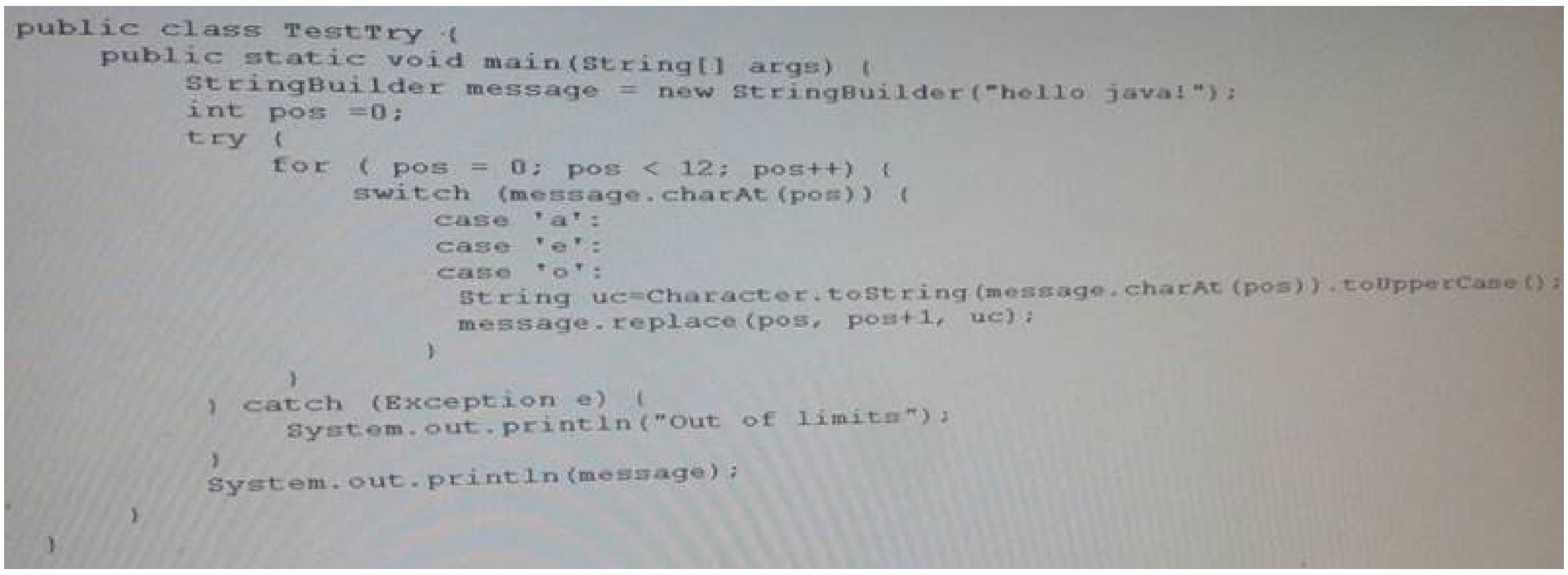


```
○ A)  for (String c:colors){
          if (c.length() != 4) {
              continue;
          }
       System.out.print(c+", ");
       }

○ B) for (String c:colors[]) {
          if (c.length() <= 4) {
              continue;
          }
      System.out.print(c+", ");
      }

○ C) for (String c:String[] colors) {
          if (c.length() >= 3) {
              continue;
          }
      System.out.print(c+", ");
      }

○ D) for (String c:colors){
          if (c.length() != 4) {
              System.out.print(c+", ");
              continue;
          }
      }
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: A

NEW QUESTION 232
Given:



```
public class TestTry {
    public static void main(String[] args) {
        StringBuilder message = new StringBuilder("hello java!");
        int pos =0;
        try {
            for ( pos = 0; pos < 12; pos++) {
                switch (message.charAt(pos)) {
                    case 'a':
                    case 'e':
                    case 'o':
                     String uc=Character.toString(message.charAt(pos)).toUpperCase();
                     message.replace(pos, pos+1, uc);
                    }
            }
        } catch (Exception e) {
            System.out.println("Out of limits");
        }
        System.out.println(message);
    }
}
```

What is the result?

A. hEllOjAvA!
B. Hello java!
C. Out of limits hEllOjAvA!
D. Out of limits

Answer: C

NEW QUESTION 234
Given the code fragment:
System.out.printIn ("Result: " +3+5); System.out.printIn ("Result: " + (3+5));
What is the result?

A. Result: 8Result: 8
B. Result: 35Result: 8
C. Result: 8Result: 35
D. Result: 35Result: 35

Answer: B

Explanation: In the first statement 3 and 5 are treated as strings andare simply concatenated. In the first statement 3 and 5 are treated as integers and their sum is calculated.

NEW QUESTION 239
Given:
public class String1 {
public static void main(String[] args) { String s = "123";
if (s.length() >2)

A. s.concat("456");for(int x = 0; x <3; x++) s += "x";System.out.println(s);}}What is the result?
B. 123
C. 123xxx
D. 123456
E. 123456xxx
F. Compilation fails

Answer: B

Explanation: 123xxx
The if clause is not applied. Note: Syntax of if-statement:
if (Statement) {
}

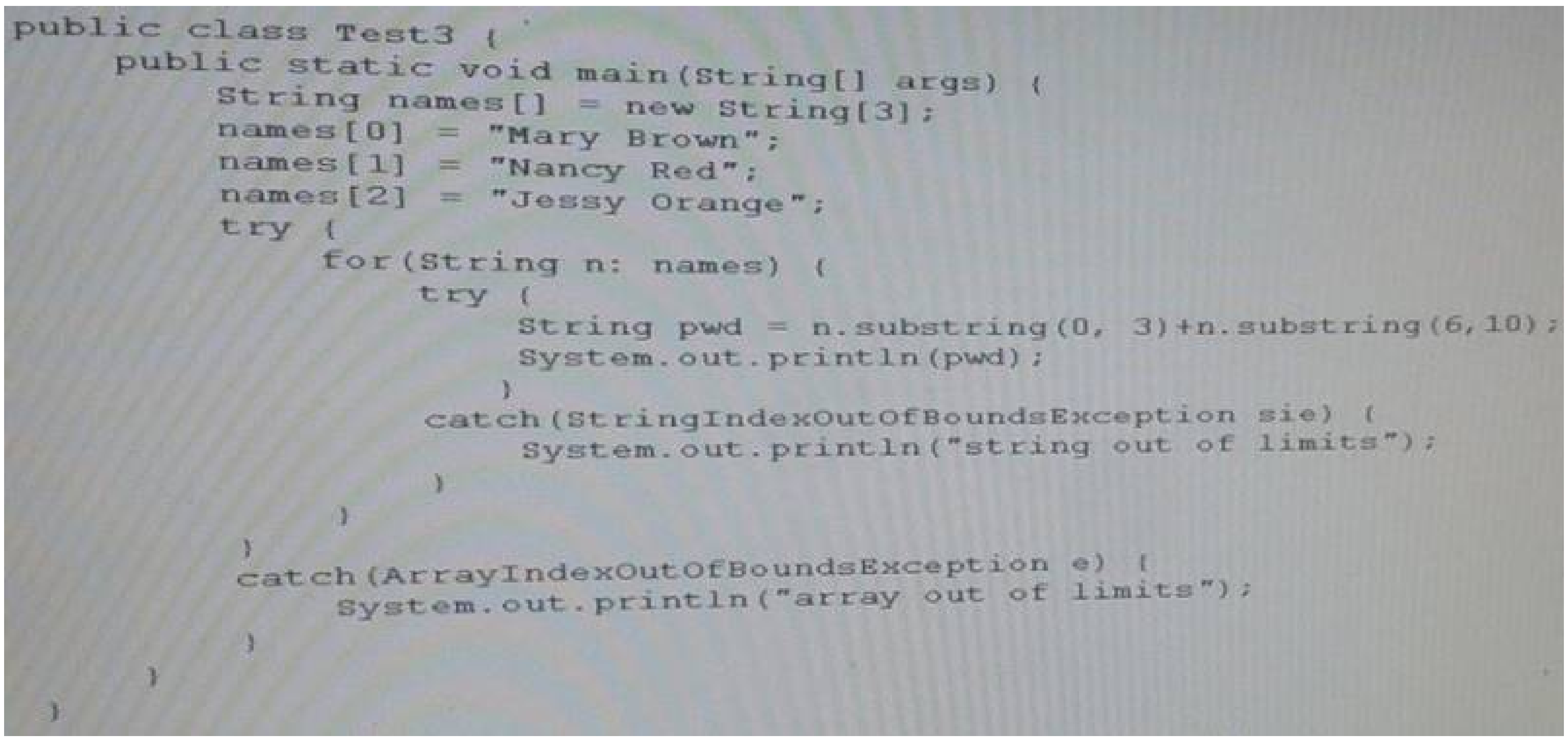
NEW QUESTION 244
Given the fragment:
String[][] arra = new String[3][]; arra[0] = new String[]{"rose", "lily"};
arra[1] = new String[]{"apple", "berry","cherry","grapes"};
arra[0] = new String[]{"beans", "carrot","potato"};
// insert code fragment here
Which code fragment when inserted at line '// insert code fragment here', enables thecode to successfully change arra elements to uppercase?

A. String[][] arra = new String[3][];arra[0] = new String[]{"rose", "lily"};arra[1] = new String[]{"apple", "berry","cherry","grapes"};arra[0] = new String[]{"beans", "carrot","potato"}; for (int i = 0; i < arra.length; i++) {for (int j=0; j < arra[i].length; j++) { arra[i][j] = arra[i][j].toUpperCase();}}
B. for (int i = 0; i < 3; i++) { for (int j=0; j< 4; j++) {arra[i][j] = arra[i][j].toUpperCase();}}
C. for (String a[]:arra[][]) { for (String x:a[]) {
D. toUpperCase();}}
E. for (int i:arra.length) { for (String x:arra) { arra[i].toUpperCase();}}

Answer: C

Explanation: Incorrect:
not A: arra.length is 3, but the subarrays have 2, 3 and 4 elements. Index will be out of bound.
not B: The subarrys are of different lengths. Indexwill be out of bound. not D: Compile error.

NEW QUESTION 247
Given:



```
public class Test3 {
    public static void main(String[] args) {
        String names[] = new String[3];
        names[0] = "Mary Brown";
        names[1] = "Nancy Red";
        names[2] = "Jessy Orange";
        try {
            for(String n: names) {
                try {
                    String pwd = n.substring(0, 3)+n.substring(6,10);
                    System.out.println(pwd);
                }
                catch(StringIndexOutOfBoundsException sie) {
                    System.out.println("string out of limits");
                }
            }
        }
        catch(ArrayIndexOutOfBoundsException e) {
            System.out.println("array out of limits");
        }
    }
}
```

What is the result?

A. Marrown String out of limits JesOran
B. Marrown String out of limits Array out of limits
C. Marrown String out of limits
D. Marrown NanRed JesOran

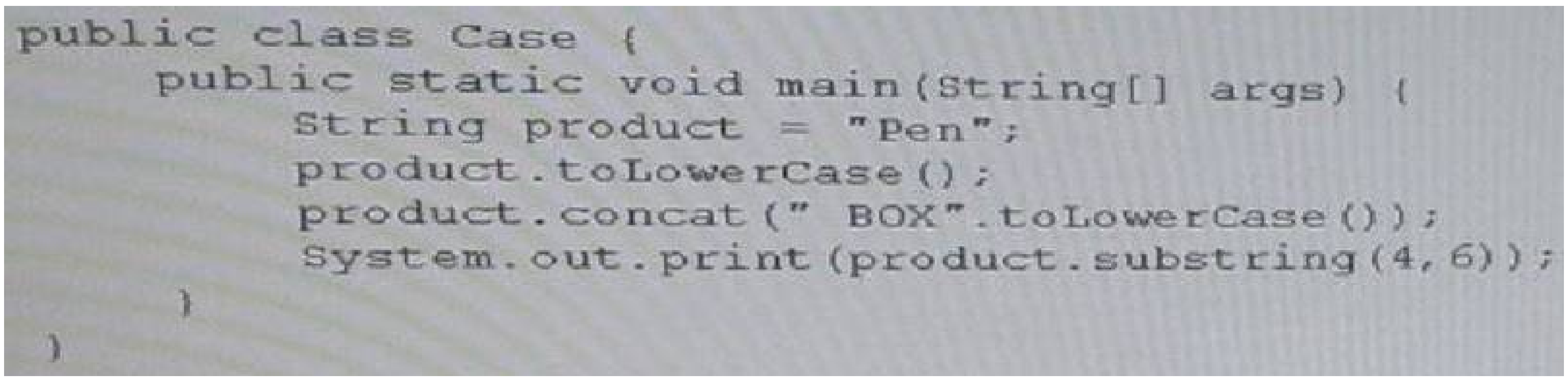
Answer: A

NEW QUESTION 250
Given the code fragment:
int b = 3;
if (!(b > 3)) {
System.out.println("square ");
}{
System.out.println("circle ");
}
System.out.println("..."); What is the result?

A. square...
B. circle...
C. squarecircle...
D. Compilation fails.

Answer: C

NEW QUESTION 252
Given:



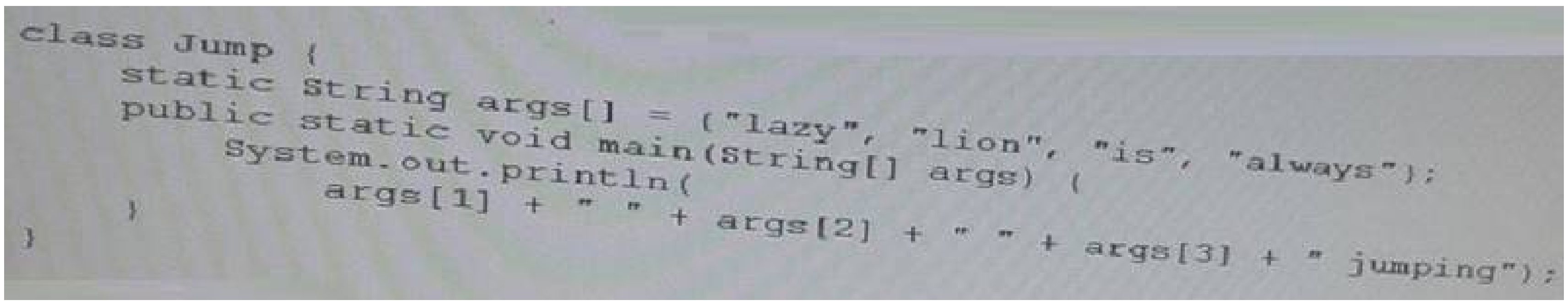
```
public class Case {
    public static void main(String[] args) {
        String product = "Pen";
        product.toLowerCase();
        product.concat(" BOX".toLowerCase());
        System.out.print(product.substring(4, 6));
    }
}
```

What is the result?

A. box
B. nbo
C. bo
D. nb
E. An exception is thrown at runtime

Answer: E

NEW QUESTION 253
Given:



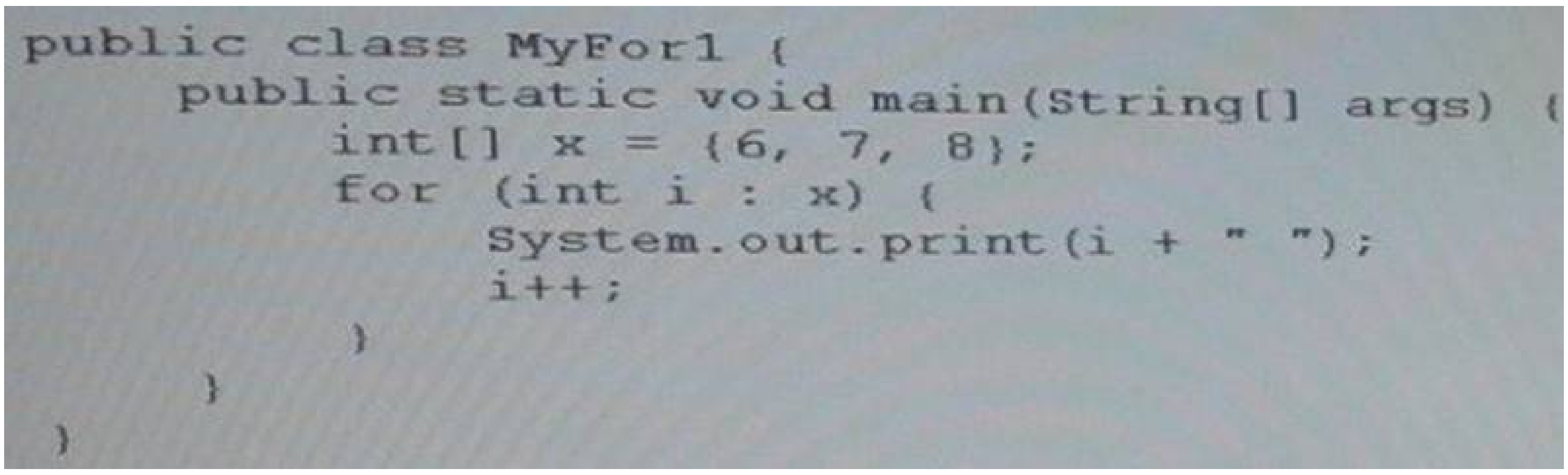
```
class Jump {
    static String args[] = {"lazy", "lion", "is", "always"};
    public static void main(String[] args) {
        System.out.println(
            args[1] + " " + args[2] + " " + args[3] + " jumping");
    }
}
```

And the commands: Javac Jump.java
Java Jump crazy elephant is always What is the result?

A. Lazy lion is jumping
B. Lion is always jumping
C. Crazy elephant is jumping
D. Elephant is always jumping
E. Compilation fails

Answer: D

NEW QUESTION 256
Given:



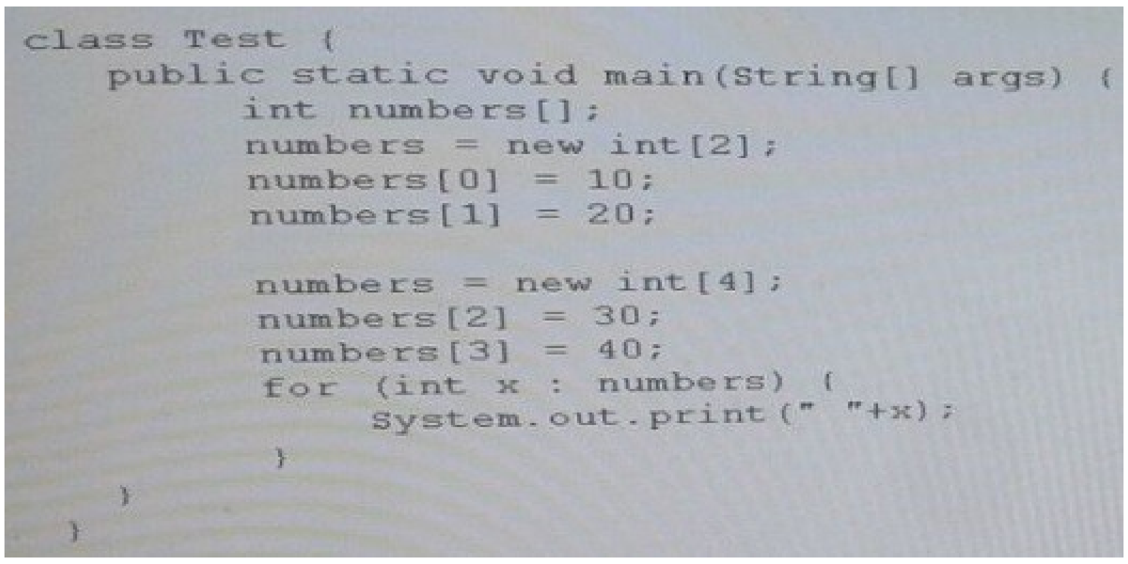
```
public class MyFor1 {
    public static void main(String[] args) {
        int[] x = {6, 7, 8};
        for (int i : x) {
            System.out.print(i + " ");
            i++;
        }
    }
}
```

What is the result?

A. 6 7 8
B. 7 8 9
C. 0 1 2
D. 6 8 10
E. Compilation fails

Answer: A

NEW QUESTION 260
Given:



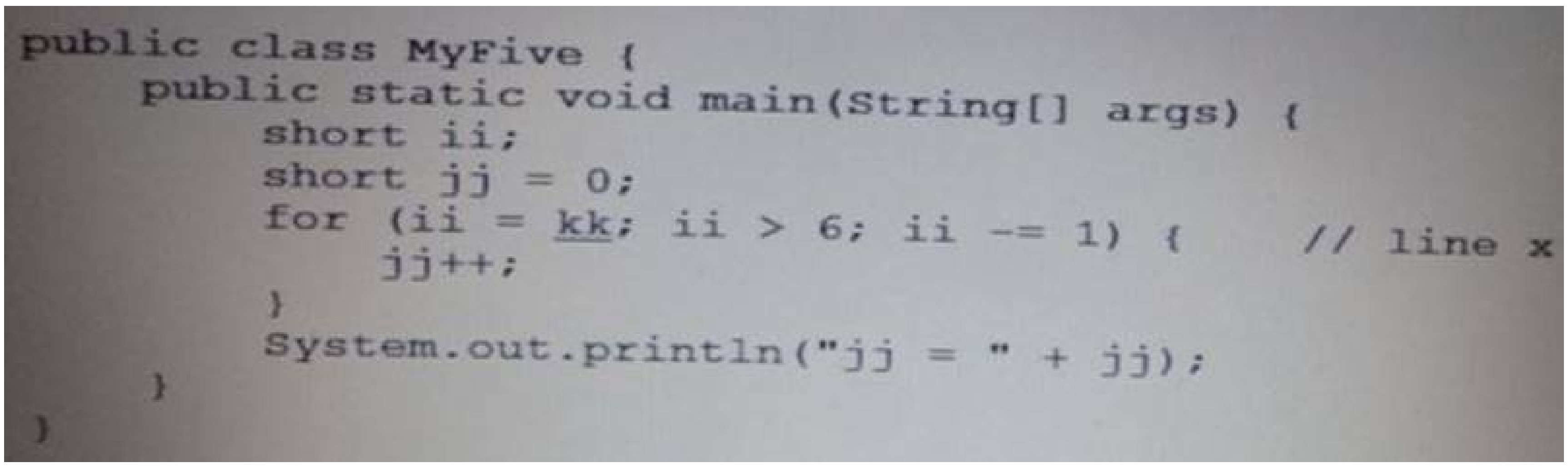
```
class Test {
    public static void main(String[] args) {
        int numbers[];
        numbers = new int[2];
        numbers[0] = 10;
        numbers[1] = 20;

        numbers = new int[4];
        numbers[2] = 30;
        numbers[3] = 40;
        for (int x : numbers) {
            System.out.print(" "+x);
        }
    }
}
```

What is the result?

A. 10 20 30 40
B. 0 0 30 40
C. Compilation fails
D. An exception is thrown at runtime

Answer: A

NEW QUESTION 262
Give:



```
public class MyFive {
    public static void main(String[] args) {
        short ii;
        short jj = 0;
        for (ii = kk; ii > 6; ii -= 1) {     // line x
            jj++;
        }
        System.out.println("jj = " + jj);
    }
}
```

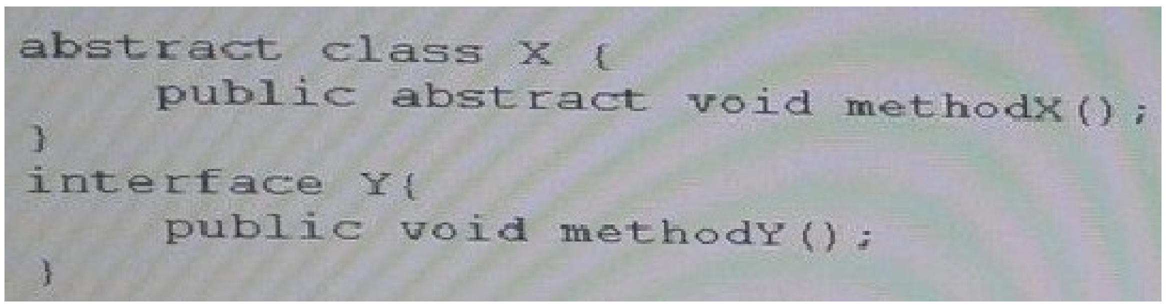
What value should replacekkin line x to cause jj = 5 to be output?

A. -1
B. 1
C. 5
D. 8
E. 11

Answer: E

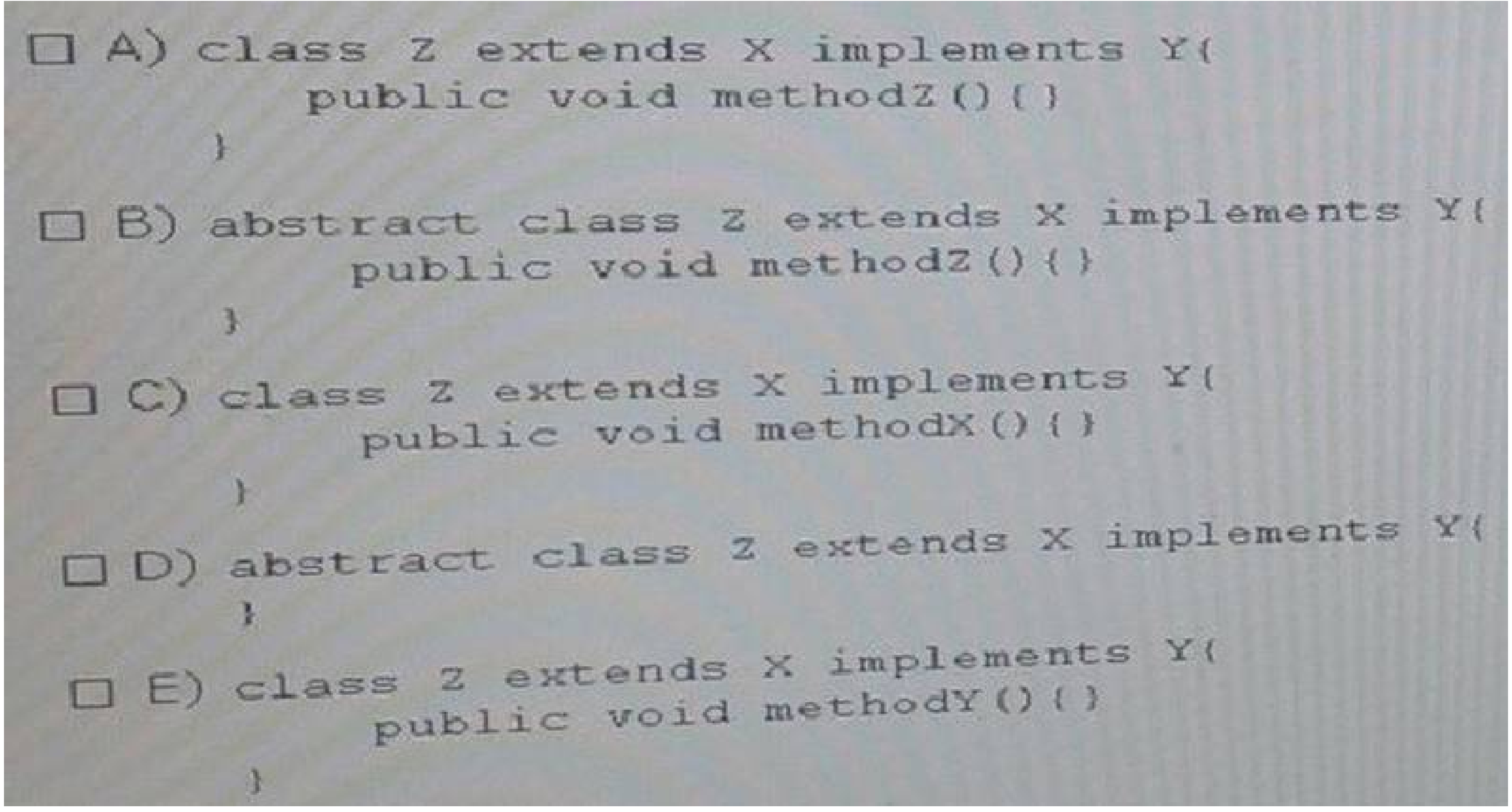
Explanation: We need to get jj to 5. It is initially set to 0. So we need to go through the for loop 5 times. The for loops ends when ii > 6 and ii decreases for every loop. So we need to initially set ii to 11. We set kk to 11.

NEW QUESTION 266
Given:



```
abstract class X {
    public abstract void methodX();
}
interface Y{
    public void methodY();
}
```

Which two code fragments are valid?



```
□ A) class Z extends X implements Y{
        public void methodZ(){}
    }

□ B) abstract class Z extends X implements Y{
        public void methodZ(){}
    }

□ C) class Z extends X implements Y{
        public void methodX(){}
    }

□ D) abstract class Z extends X implements Y{
    }

□ E) class Z extends X implements Y{
        public void methodY(){}
    }
```

A. Option A
B. Option B
C. Option C
D. Option D
E. Option E

Answer: BC

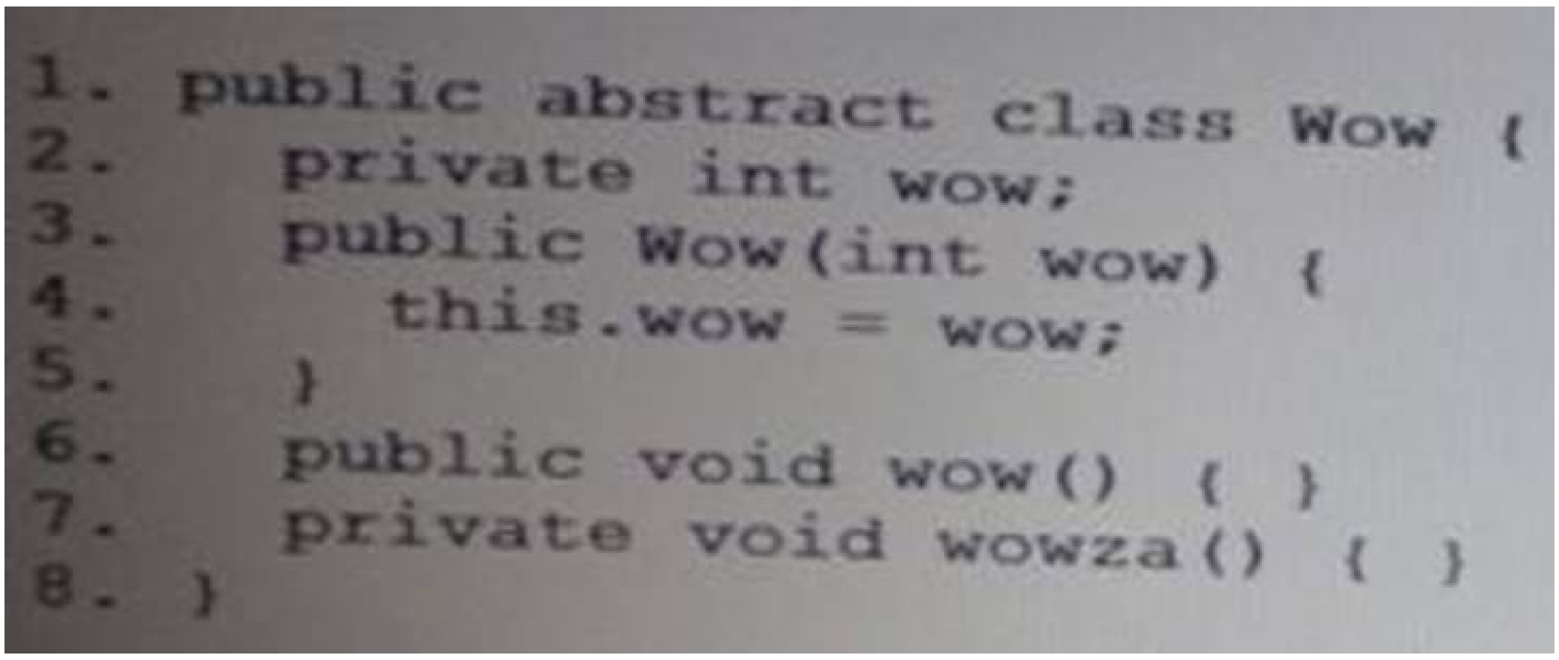
Explanation: When an abstract class is subclassed, the subclass usually provides implementations for all of the abstract methods in its parent class (C). However, if it does not, thenthe subclass must also be declared abstract (B).
Note: An abstract class is a class that is declared abstract—it may or may not include abstract methods. Abstract classes cannot be instantiated, but they can be subclassed.

NEW QUESTION 267
Given:
public class ScopeTest { int j, int k;
public static void main(String[] args) { ew ScopeTest().doStuff(); }
void doStuff() { nt x = 5; oStuff2();
System.out.println("x");
}
voiddoStuff2() { nt y = 7;
ystem.out.println("y");
or (int z = 0; z < 5; z++) { ystem.out.println("z");
ystem.out.println("y");
}
Which two items are fields?

A. j
B. k
C. x
D. y
E. z

Answer: AB

NEW QUESTION 268
Given:



```
public abstract class Wow {
    private int wow;
    public Wow(int wow) {
        this.wow = wow;
    }
    public void wow() { }
    private void wowza() { }
}
```

What is true about the class Wow?

A. It compiles without error.
B. It does not compile because an abstract class cannot have private methods.
C. It does not compile because an abstract class cannot have instance variables.
D. It does not compile because an abstract class must have at least one abstract method.
E. It does not compile because an abstract class must have a constructor with no arguments.

Answer: A

NEW QUESTION 270
Given:
class X {}
class Y { Y ( ) { } }
class Z { Z (int i) { } }
Which class has a default constructor?

A. X only
B. Y only
C. Z only
D. X and Y
E. Y and Z
F. X and Z
G. X, Y and Z

Answer: A

NEW QUESTION 271
Given the code fragment:
Int [] [] array = {{0}, {0, 1}, {0, 2, 4}, {0, 3, 6, 9}, {0, 4, 8, 12, 16}};
Systemout.printIn(array [4] [1]);
System.out.printIn (array) [1] [4]); What is the result?

A. 4Null
B. Null 4
C. An IllegalArgumentException is thrown at run time

D. 4An ArrayIndexOutOfBoundException is thrown at run time

Answer: D

Explanation: The first println statement, System.out.println(array [4][1]);, works fine. It selects the element/array with index 4, {0, 4, 8, 12, 16},and from this array it selects the element with index 1, 4. Output: 4
The second println statement, System.out.println(array) [1][4]);, fails. It selects the array/element with index 1, {0, 1}, and from this array it try to select the element with index
4. This causes an exception.
Output: 4
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: 4

NEW QUESTION 275
Given:

```
public class MyClass {
public static void main(String[] args) { String s = " Java Duke ";
int len = s.trim().length(); System.out.print(len);
}
}
```

What is the result?

A. 8
B. 9
C. 11
D. 10
E. Compilation fails

Answer: B

Explanation: Java -String trim() Method
This method returns a copy of the string, with leading and trailing whitespace omitted.

NEW QUESTION 279
Given:

```
public class Test {
public static void main(String[] args) { int day = 1;
switch (day) {
case "7": System.out.print("Uranus");
case "6": System.out.print("Saturn");
case "1": System.out.print("Mercury");
case "2": System.out.print("Venus");
case "3": System.out.print("Earth");
case "4": System.out.print("Mars");
case "5": System.out.print("Jupiter");
}
}
}
```

Which two modifications, made independently, enable the code to compile and run?

A. Adding a break statement after each print statement
B. Adding a default section within the switch code-block
C. Changing the string literals in each case label to integer
D. Changing the type of the variable day to String
E. Arranging the case labels in ascending order

Answer: AC

Explanation: The following will work fine:

```
public class Test {
public static void main(String[] args) { int day = 1;
switch (day) {
case 7: System.out.print("Uranus"); break; case 6:System.out.print("Saturn"); break; case 1: System.out.print("Mercury"); break; case 2: System.out.print("Venus"); break; case 3: System.out.print("Earth"); break; case 4: System.out.print("Mars"); break; case 5: System.out.print("Jupiter"); break;
}
}
}
```

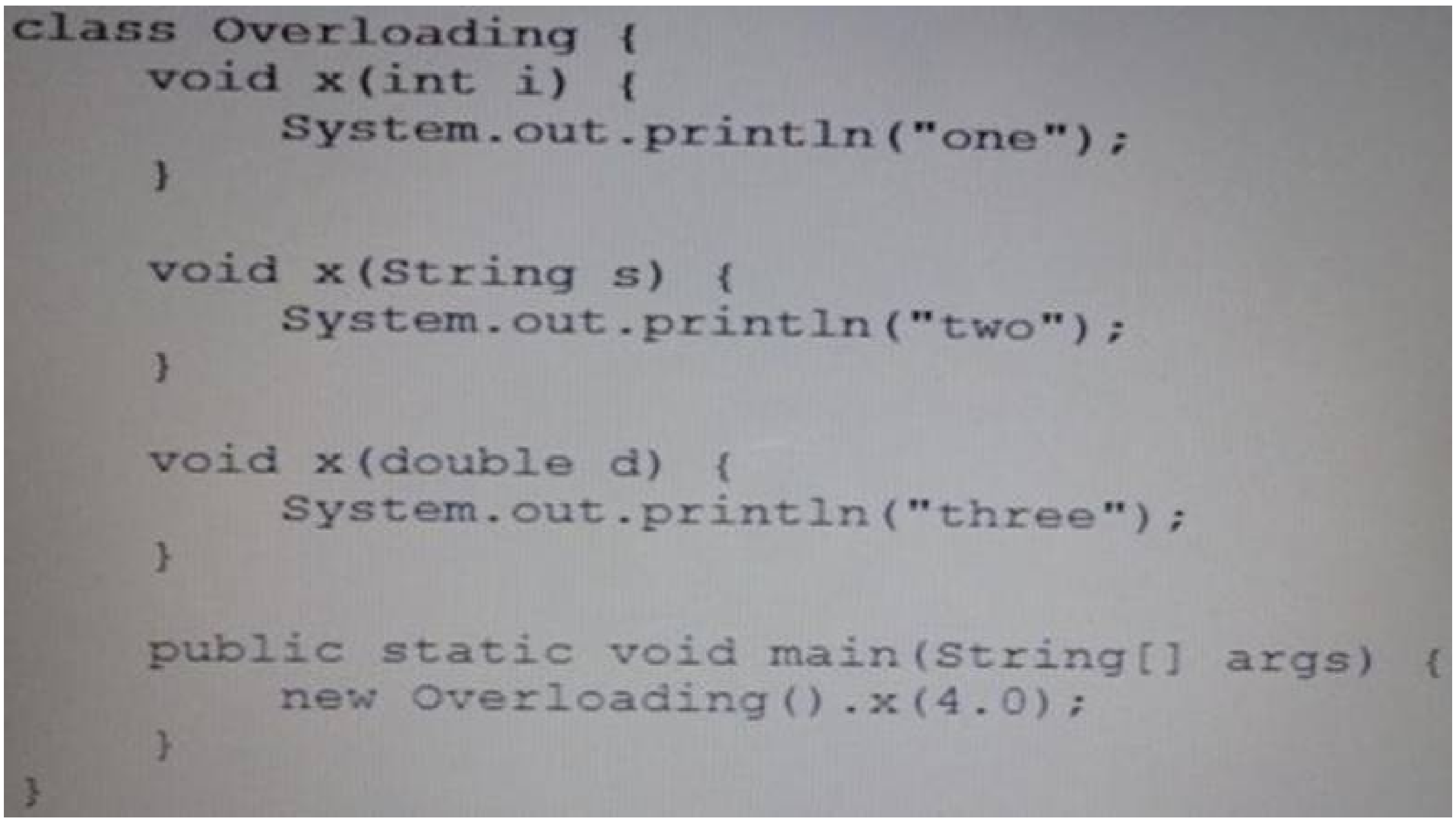
NEW QUESTION 282
A method doSomething () that has no exception handling code is modified to trail a method that throws achecked exception. Which two modifications, made independently, will allow the program to compile?

A. Catch the exception in the method doSomething().
B. Declare the exception to be thrown in the doSomething() method signature.
C. Cast the exception to aRunTimeException in the doSomething() method.
D. Catch the exception in the method that calls doSomething().

Answer: AB

Explanation: Valid Java programming language code must honor the Catch or Specify Requirement. This means that code that might throw certain exceptions must be enclosed by either of the following:
* A try statement that catches the exception. The try must provide a handler for the exception, as described in Catching and Handling Exceptions.
* A method that specifies that it can throw the exception. The method must provide a throws clause that lists the exception, as described in Specifying the Exceptions Thrown by a Method.
Code that fails to honor the Catch or Specify Requirement will not compile.

NEW QUESTION 286
Given:



```
class Overloading {
     void x(int i) {
          System.out.println("one");
     }

     void x(String s) {
          System.out.println("two");
     }

     void x(double d) {
          System.out.println("three");
     }

     public static void main(String[] args) {
          new Overloading().x(4.0);
     }
}
```

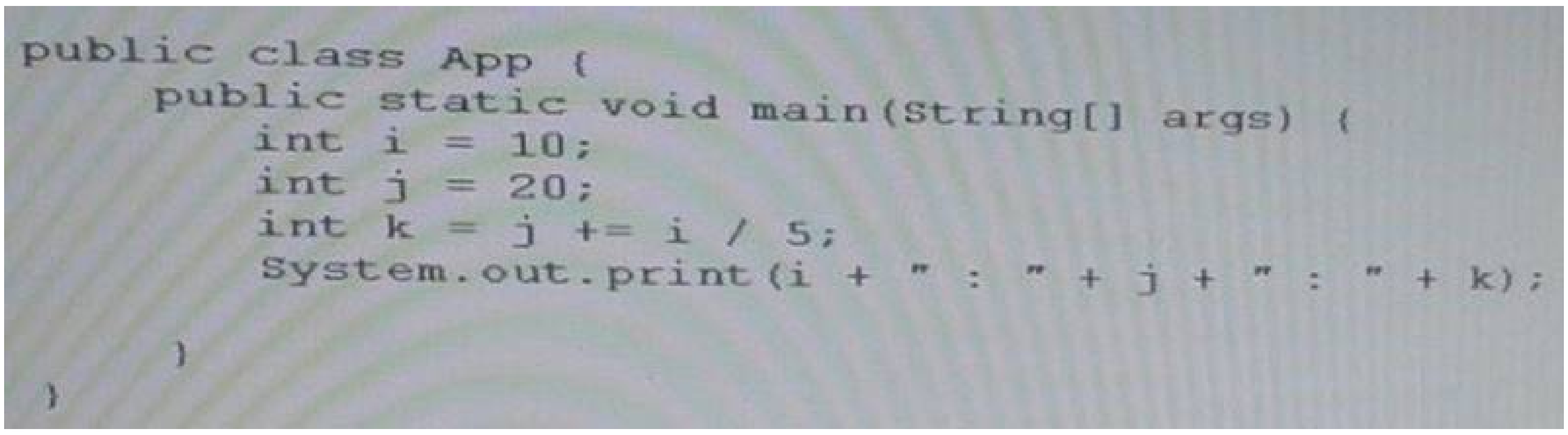
What is the result?

A. One
B. Two
C. Three
D. Compilation fails

Answer: C

Explanation: In this scenario the overloading method is called with a double/float value, 4.0. This makes the third overload method to run.
Note:
The Java programming language supportsoverloadingmethods, and Java can distinguish between methods with differentmethod signatures. This means that methods within a class can have the same name if they have different parameter lists. Overloaded methods are differentiated by the number and the type of the arguments passed into the method.

NEW QUESTION 291
Given:



```
public class App {
     public static void main(String[] args) {
          int i = 10;
          int j = 20;
          int k = j += i / 5;
          System.out.print(i + " : " + j + " : " + k);
     }
}
```

What is the result?

A. 10 : 22 : 20
B. 10 : 22 : 22
C. 10 : 22 : 6
D. 10 : 30 : 6

Answer: B

NEW QUESTION 296
Given the code fragment:

int a = 0; a++;
System.out.printIn(a++); System.out.printIn(a);
What is the result?

A. 12
B. 01
C. 11
D. 22

Answer: A

Explanation: The first println prints variable a with value 1 and then increases the variable to 2.

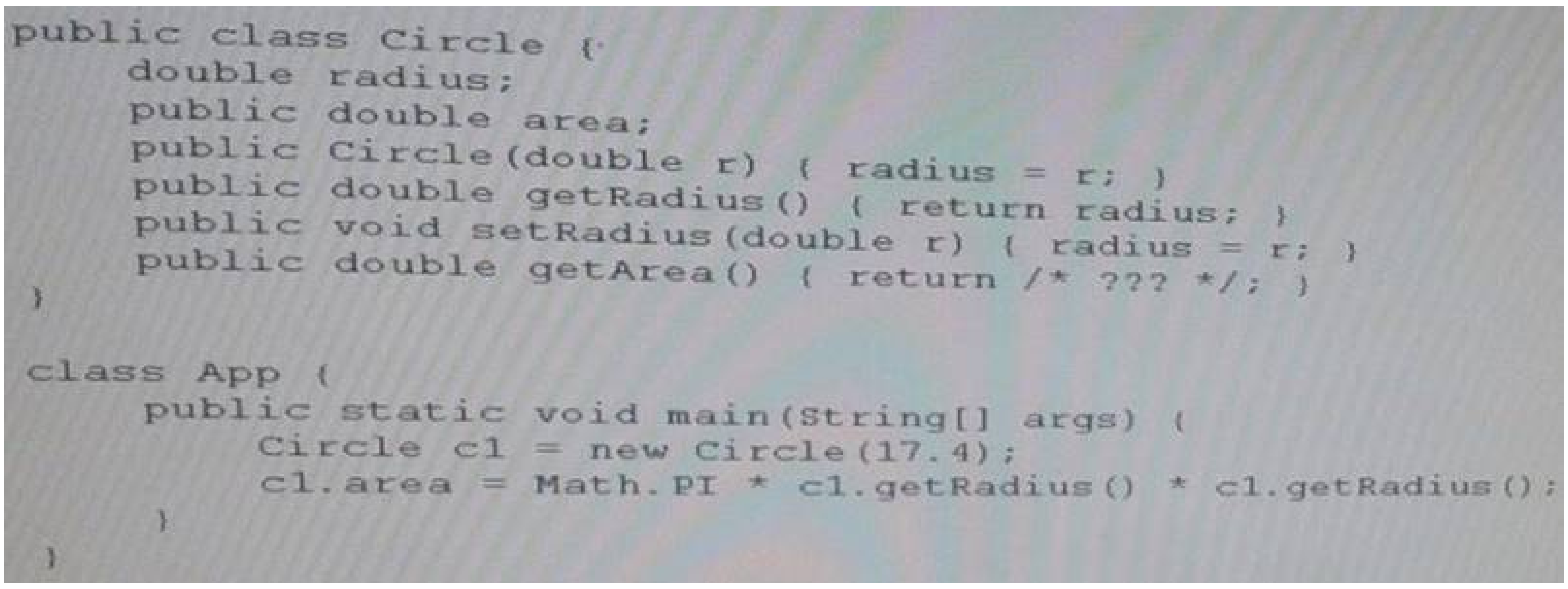
NEW QUESTION 297
Given the code fragment:
public class Test {
static String[][] arr =new String[3][]; private static void doPrint() {
//insert code here
}
public static void main(String[] args) { String[] class1 = {"A","B","C"};
String[] class2 = {"L","M","N","O"}; String[] class3 = {"I","J"};
arr[0] = class1; arr[1] = class2; arr[2] = class3; Test.doPrint();
}
}
Which code fragment, when inserted at line //insert code here, enables the code to print COJ?

A. int i = 0;for (String[] sub: arr) { int j = sub.length -1; for (String str: sub) {System.out.println(str[j]); i++;}}
B. private static void doPrint() { for (int i = 0;i < arr.length;i++) { int j = arr[i].length-1; System.out.print(arr[i][j]);}}
C. int i = 0;for (String[] sub: arr[][]) { int j = sub.length; System.out.print(arr[i][j]); i++;}
D. for (int i = 0;i < arr.length-1;i++) { int j = arr[i].length-1;System.out.print(arr[i][j]); i++;}

Answer: B

Explanation: Incorrect:
not A: The following line causes a compile error: System.out.println(str[j]);
Not C: Compile erro line: for (String[] sub: arr[][]) not D: Output: C

NEW QUESTION 301
Given:



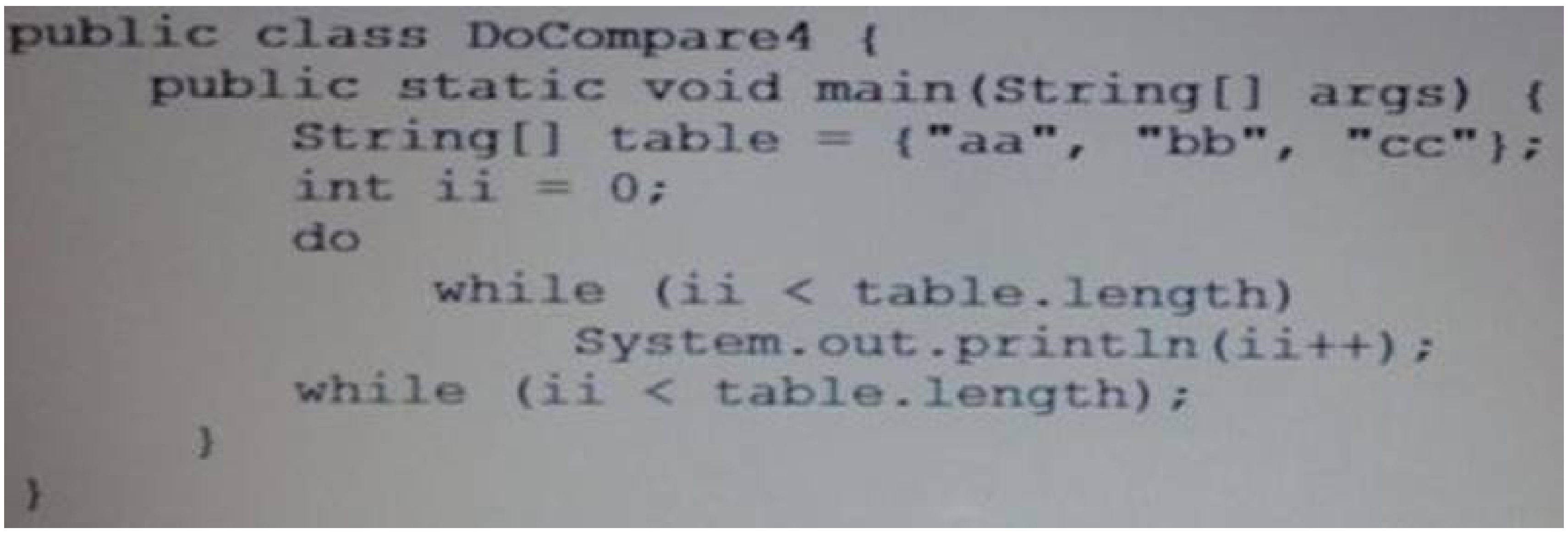
```
public class Circle {
    double radius;
    public double area;
    public Circle(double r) { radius = r; }
    public double getRadius() { return radius; }
    public void setRadius(double r) { radius = r; }
    public double getArea() { return /* ??? */; }
}

class App {
    public static void main(String[] args) {
        Circle c1 = new Circle(17.4);
        c1.area = Math.PI * c1.getRadius() * c1.getRadius();
    }
}
```

The class is poorly encapsulated. You need to change the circle class to compute and return the area instead.
Which two modifications are necessary to ensure that the class is being properly encapsulated?

A. Remove the area field.
B. Change the getArea( ) method as follows:public double getArea ( ) { return Match.PI * radius * radius; }
C. Add the following method:public double getArea ( ) {area = Match.PI * radius * radius; }
D. Change the cacess modifier of the SerRadius ( ) method to be protected.

Answer: BD

NEW QUESTION 303
Given:



```
public class DoCompare4 {
    public static void main(String[] args) {
        String[] table = {"aa", "bb", "cc"};
        int ii = 0;
        do
            while (ii < table.length)
                System.out.println(ii++);
        while (ii < table.length);
    }
}
```

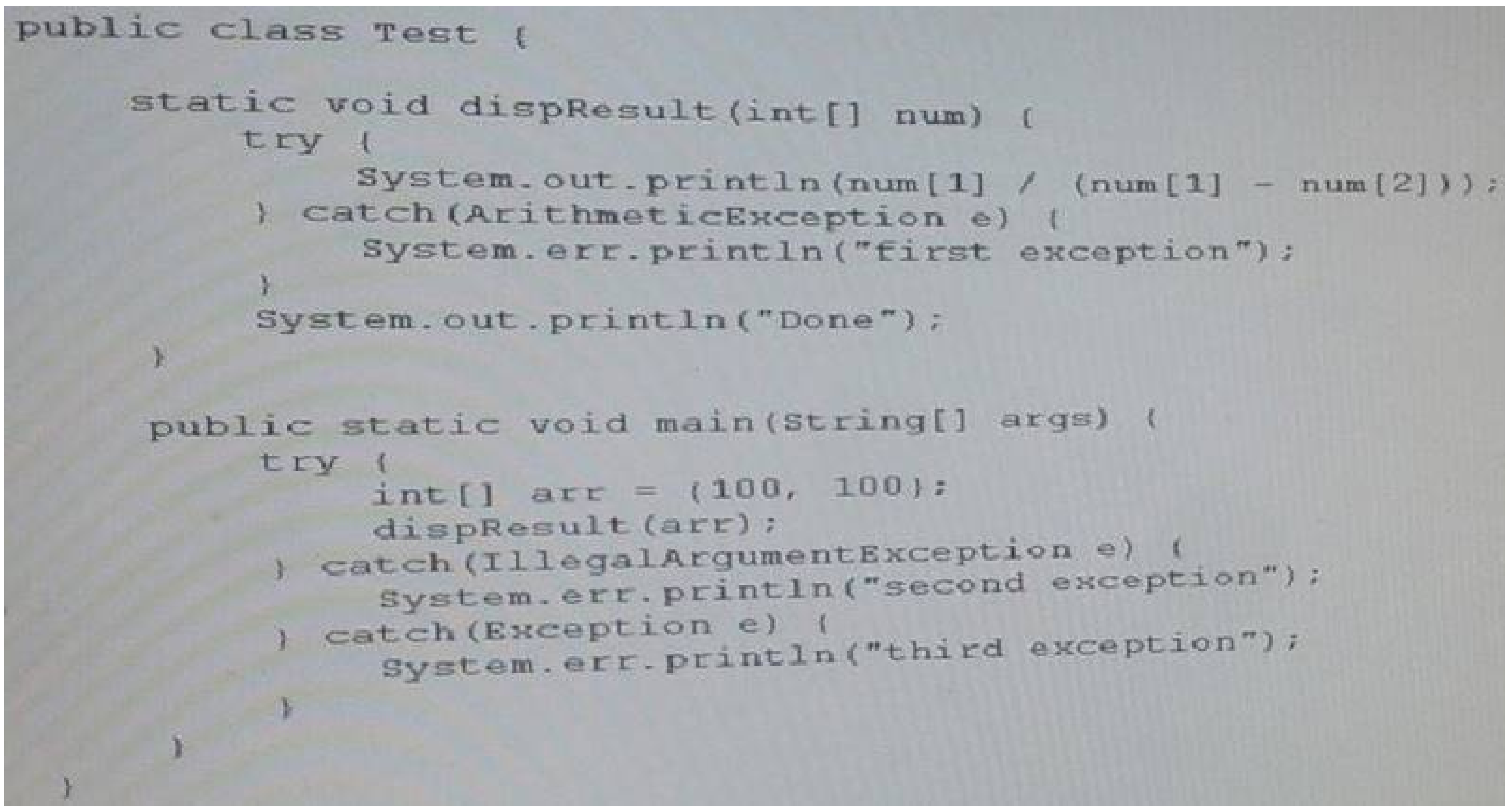
What is the result?

A. 012
B. 012012012
C. Compilation fails

Answer: B

Explanation: table.length is 3. So the do-while loop will run 3 times with ii=0, ii=1 and ii=2. The second while statement will break the do-loop when ii = 3.
Note:The Java programming language provides ado-whilestatement, which can be expressed as follows:
do { statement(s)
} while (expression);

NEW QUESTION 307
Given:



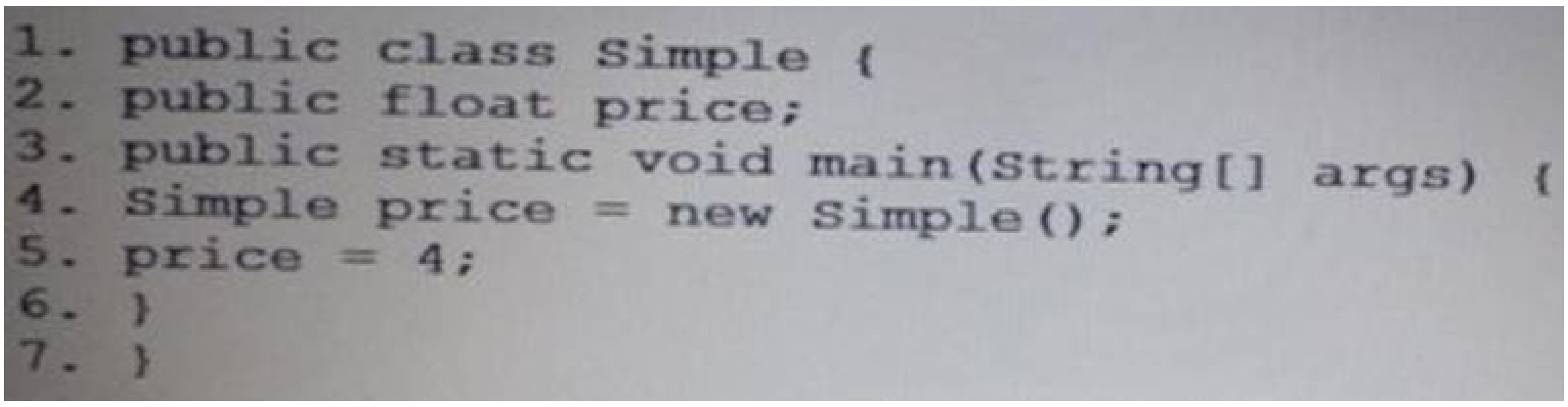
```
public class Test {

    static void dispResult(int[] num) {
        try {
            System.out.println(num[1] / (num[1] - num[2]));
        } catch(ArithmeticException e) {
            System.err.println("first exception");
        }
        System.out.println("Done");
    }

    public static void main(String[] args) {
        try {
            int[] arr = {100, 100};
            dispResult(arr);
        } catch(IllegalArgumentException e) {
            System.err.println("second exception");
        } catch(Exception e) {
            System.err.println("third exception");
        }
    }
}
```

What is the result?

A. 0Done
B. First Exception Done
C. Second Exception
D. DoneThird Exception
E. Third Exception

Answer: B

NEW QUESTION 310
Given the following code:



```
1. public class Simple {
2. public float price;
3. public static void main(String[] args) {
4. Simple price = new Simple();
5. price = 4;
6. }
7. }
```

What will make this code compile and run?

A. Change line 2 to the following: Publicint price
B. Change line 4 to the following: int price = new simple ();
C. Change line 4 to the following: Float price = new simple ();
D. Change line 5 to the following: Price = 4f;
E. Change line 5 to the following: price.price = 4;
F. Change line 5to the following: Price = (float) 4:
G. Change line 5 to the following: Price = (Simple) 4;
H. The code compiles and runs properly; no changes are necessary

Answer: E

Explanation: price.price =4; is correct, not price=4;
The attribute price of the instance must be set, not the instance itself.

NEW QUESTION 315
Given the classes:
* AssertionError
* ArithmeticException
* ArrayIndexOutofBoundsException
* FileNotFoundException
* IllegalArgumentException
* IOError
* IOException
* NumberFormatException
* SQLException
Which option lists only those classes that belong to the unchecked exception category?

A. AssertionError, ArrayIndexOutOfBoundsException, ArithmeticException
B. AssertionError, IOError, IOException
C. ArithmeticException, FileNotFoundException, NumberFormatException
D. FileNotFoundException, IOException, SQLException
E. ArrayIndexOutOfBoundException, IllegalArgumentException, FileNotFoundException

Answer: A

Explanation: Not B: IOError and IOException are both checked errors. Not C, not D, not E: FileNotFoundException is a checked error.
Note:
Checked exceptions:
* represent invalid conditions in areas outside the immediate control of the program (invalid user input, database problems, network outages, absent files)
* are subclasses of Exception
* a method is obliged to establish a policy for all checked exceptions thrown by its implementation (either pass the checked exception further up the stack, or handle it somehow)
Note:
Unchecked exceptions:
* represent defects in the program (bugs) - often invalid arguments passed to a non-private method. To quote from The Java Programming Language, by Gosling, Arnold, and Holmes: "Unchecked runtime exceptions represent conditions that, generally speaking,reflect errors in your program's logic and cannot be reasonably recovered from at run time."
* are subclasses of RuntimeException, and are usually implemented using IllegalArgumentException, NullPointerException, or IllegalStateException
* method is not obliged to establish a policy for the unchecked exceptions thrown by its implementation (and they almost always do not do so)

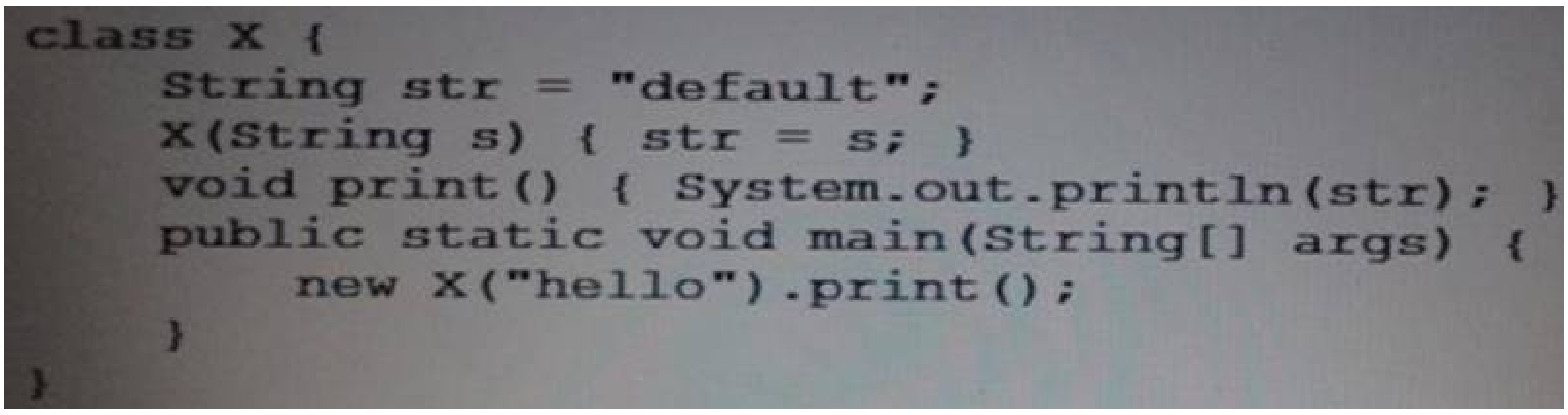
NEW QUESTION 319
Given the code fragment:
Boolean b1 = true; Boolean b2 = false; int i = 0;
while (foo) { }
Which one is valid as a replacement for foo?

A. b1.compareTo(b2)
B. i = 1
C. i == 2? -1 : 0
D. "foo".equals("bar")

Answer: D

Explanation: Equals works fine on strings equals produces a Boolean value.

NEW QUESTION 321
Given:



```
class X {
    String str = "default";
    X(String s) { str = s; }
    void print() { System.out.println(str); }
    public static void main(String[] args) {
        new X("hello").print();
    }
}
```

What is the result?

A. Hello
B. Default
C. Compilation fails
D. The program prints nothing
E. An exception is thrown at run time

Answer: A

Explanation: The program compilesfine. The program runs fine. The output is: hello

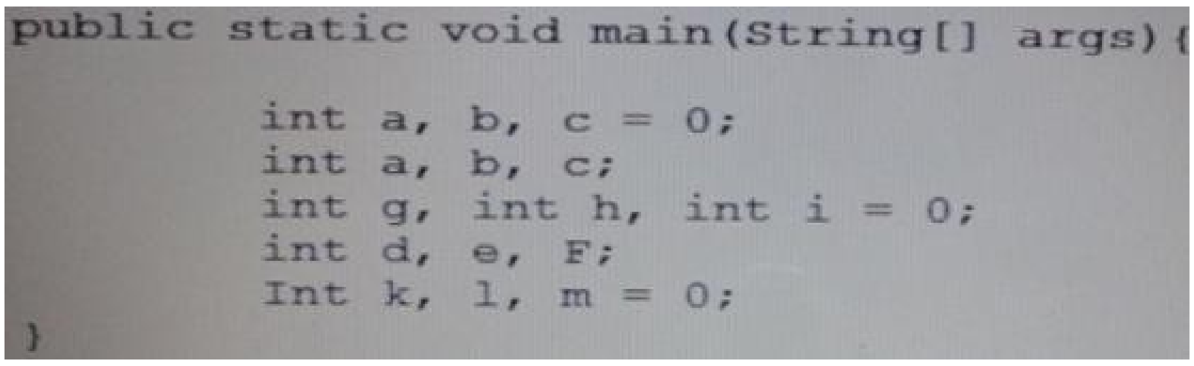
NEW QUESTION 325
Given the code fragment:
class Student {
int rollnumber; String name;
List cources = new ArrayList();
// insert code here public String toString() {
return rollnumber + " : " + name + " : " + cources;
}
}
And,
public class Test {
public static void main(String[] args) { List cs = newArrayList(); cs.add("Java");
cs.add("C");
Student s = new Student(123,"Fred", cs); System.out.println(s);
}
}
Which code fragment, when inserted at line // insert code here, enables class Test to print 123 : Fred : [Java, C]?

A. private Student(int i, String name, List cs) {/* initialization code goes here */}
B. public void Student(int i, String name, List cs) {/* initialization code goes here */}
C. Student(int i, String name, List cs) {/* initialization code goes here */}
D. Student(int i, String name, ArrayList cs) {/* initialization code goes here */}

Answer: C

Explanation: Incorrect:
Not A: Student has private access line: Student s = new Student(123,"Fred", cs);
Not D: Cannot be applied to given types.Line: Student s = new Student(123,"Fred", cs);

NEW QUESTION 330
Given:



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

        int a, b, c = 0;
        int a, b, c;
        int g, int h, int i = 0;
        int d, e, F;
        Int k, l, m = 0;
}
```

Which two declarations will compile?

A. int a, b, c = 0;
B. int a, b, c;
C. int g, int h, int i = 0;
D. int d, e, F;
E. int k, l, m; = 0;

Answer: AD

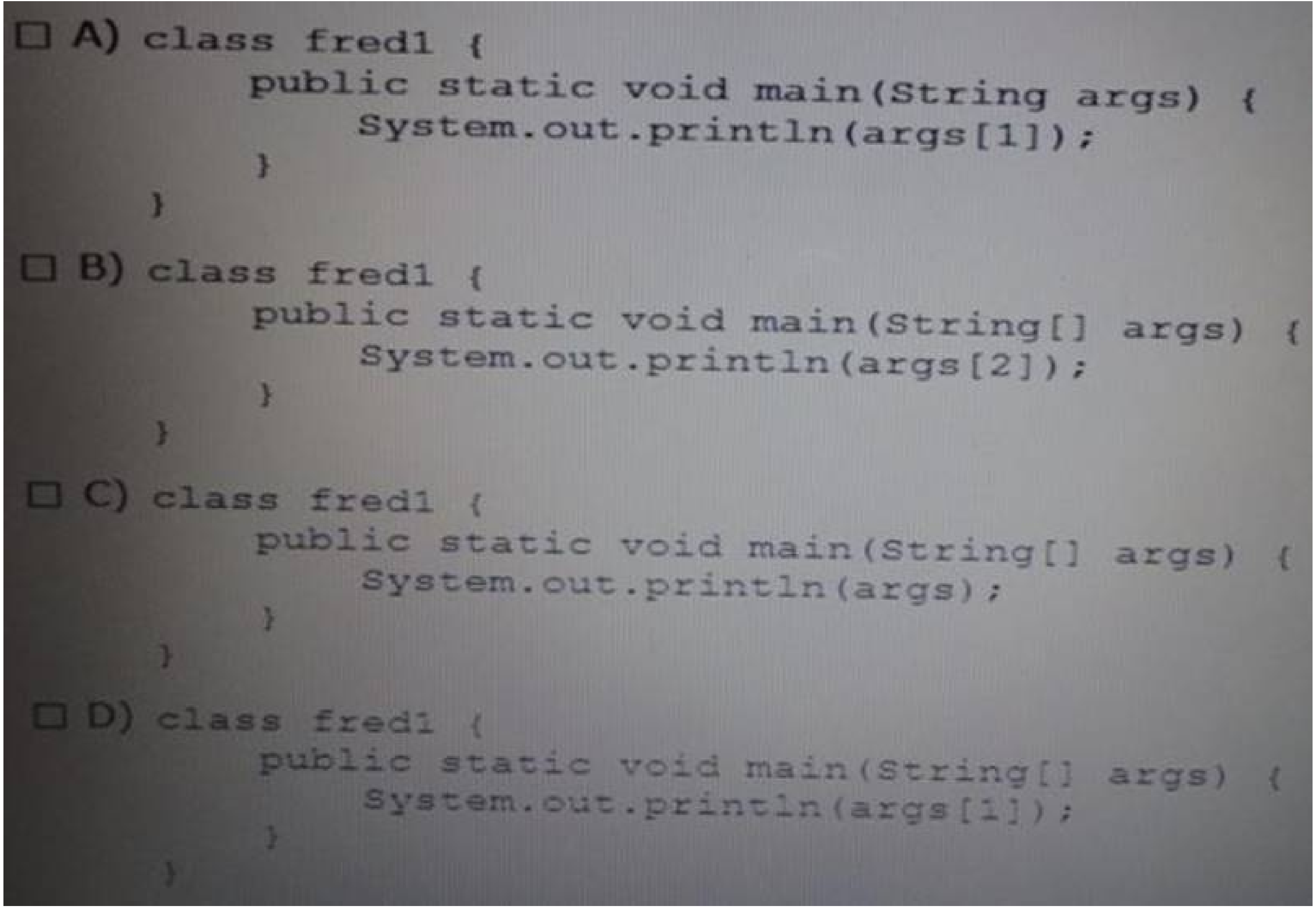
NEW QUESTION 335
Given the code fragment:
StringBuilder sb = new StringBuilder ( ) ; Sb.append (“world”);
Which code fragment prints Hello World?

A. sb.insert(0,"Hello "); System.out.println(sb);
B. sb.append(0,"Hello "); System.out.println(sb);
C. sb.add(0,"Hello "); System.out.println(sb);
D. sb.set(0,"Hello "); System.out.println(sb);D

Answer: A

Explanation: The java.lang.StringBuilder.insert(int offset, char c) method inserts the string representation of the char argument into this sequence.
The second argument is inserted into the contents of this sequence at the position indicated by offset. The length of this sequence increases by one.The offset argument must be greater than or equal to 0, and less than or equal to the length of this sequence.
Reference: Java.lang.StringBuilder.insert() Method

NEW QUESTION 338
Which two will compile, and can be run successfully using the command: Java fred1 hello walls



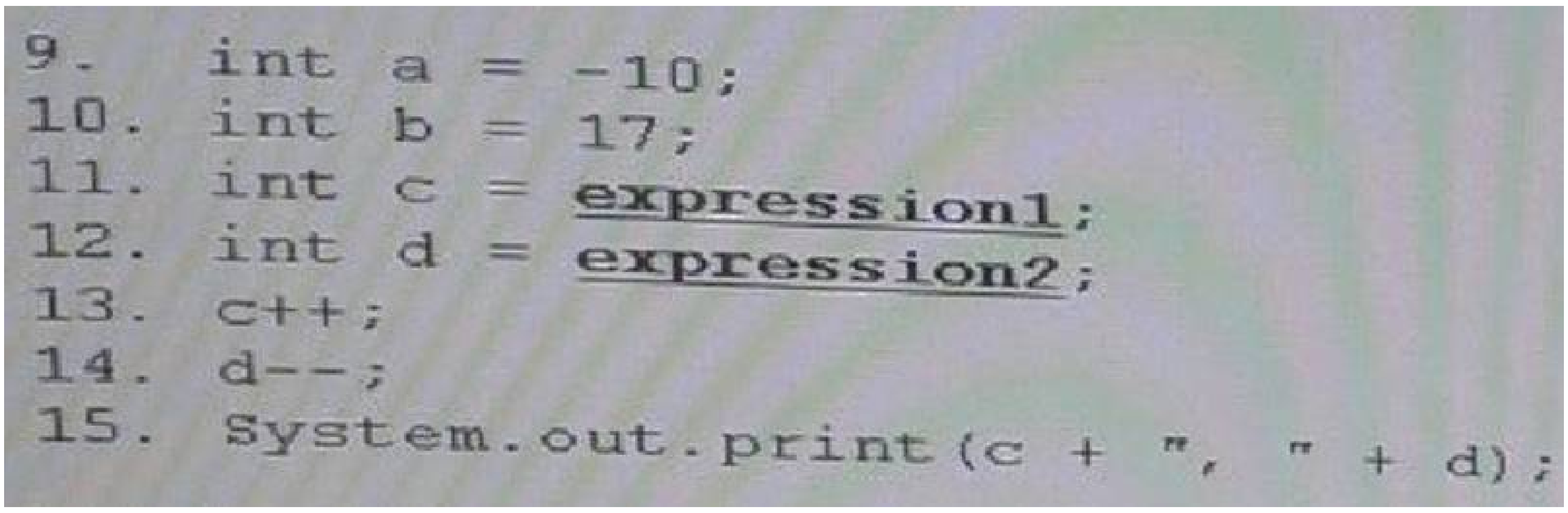
```
□ A) class fred1 {
         public static void main(String args) {
             System.out.println(args[1]);
         }
     }
□ B) class fred1 {
         public static void main(String[] args) {
             System.out.println(args[2]);
         }
     }
□ C) class fred1 {
         public static void main(String[] args) {
             System.out.println(args);
         }
     }
□ D) class fred1 {
         public static void main(String[] args) {
             System.out.println(args[1]);
         }
     }
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: CD

Explanation: Throwsjava.lang.ArrayIndexOutOfBoundsException: 2 at certquestions.Fred1.main(Fred1.java:3)
:C. Prints out: [Ljava.lang.String;@39341183
:D. Prints out: walls

NEW QUESTION 340
Given the code fragment:



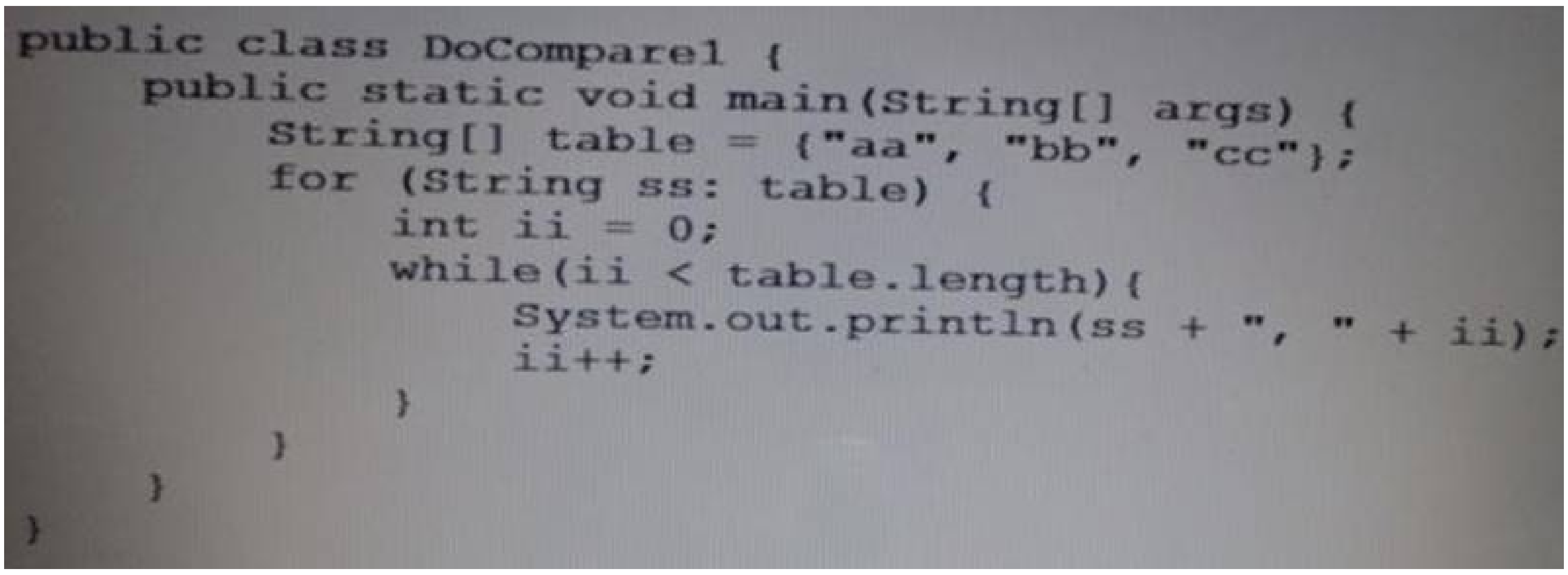
```
int a = -10;
int b = 17;
int c = expression1;
int d = expression2;
c++;
d--;
System.out.print(c + ", " + d);
```

What couldexpression1andexpression2be, respectively, in order to produce output –8, 16?

A. + +a, - -b
B. + +a, b- -
C. A+ +, - - b
D. A + +, b - -

Answer: B

NEW QUESTION 342
Given:



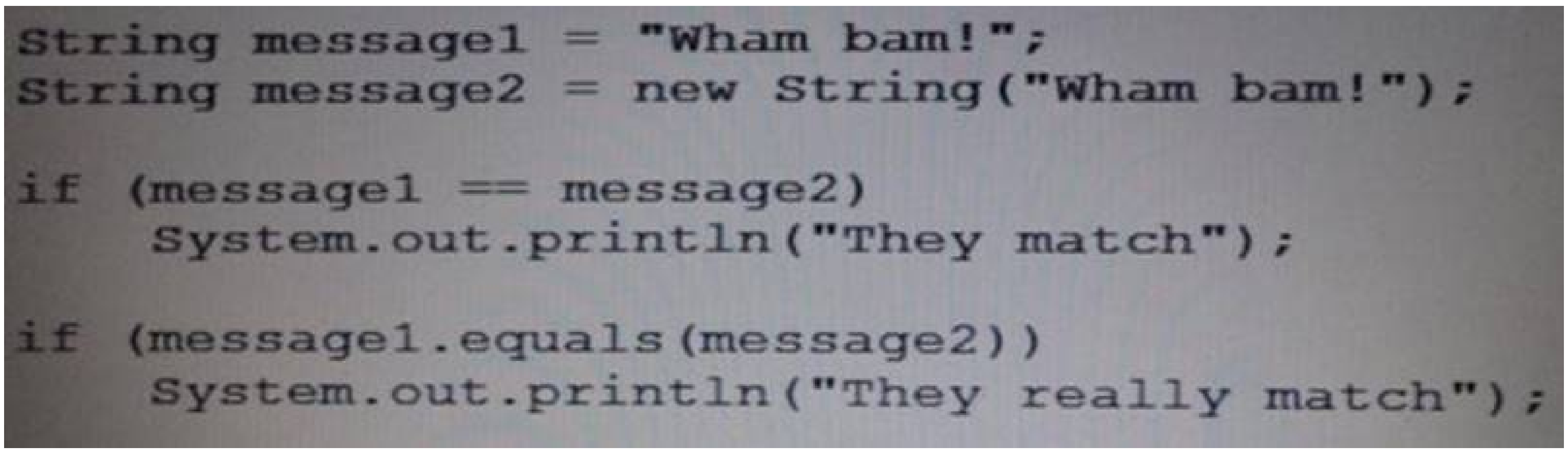
```
public class DoCompare1 {
    public static void main(String[] args) {
        String[] table = {"aa", "bb", "cc"};
        for (String ss: table) {
            int ii = 0;
            while(ii < table.length){
                System.out.println(ss + ", " + ii);
                ii++;
            }
        }
    }
}
```

How many times is 2 printed as a part of the output?

A. Zero
B. Once
C. Twice
D. Thrice
E. Compilation fails.

Answer: A

NEW QUESTION 345
Given:



```
String message1 = "Wham bam!";
String message2 = new String("Wham bam!");

if (message1 == message2)
    System.out.println("They match");

if (message1.equals(message2))
    System.out.println("They really match");
```

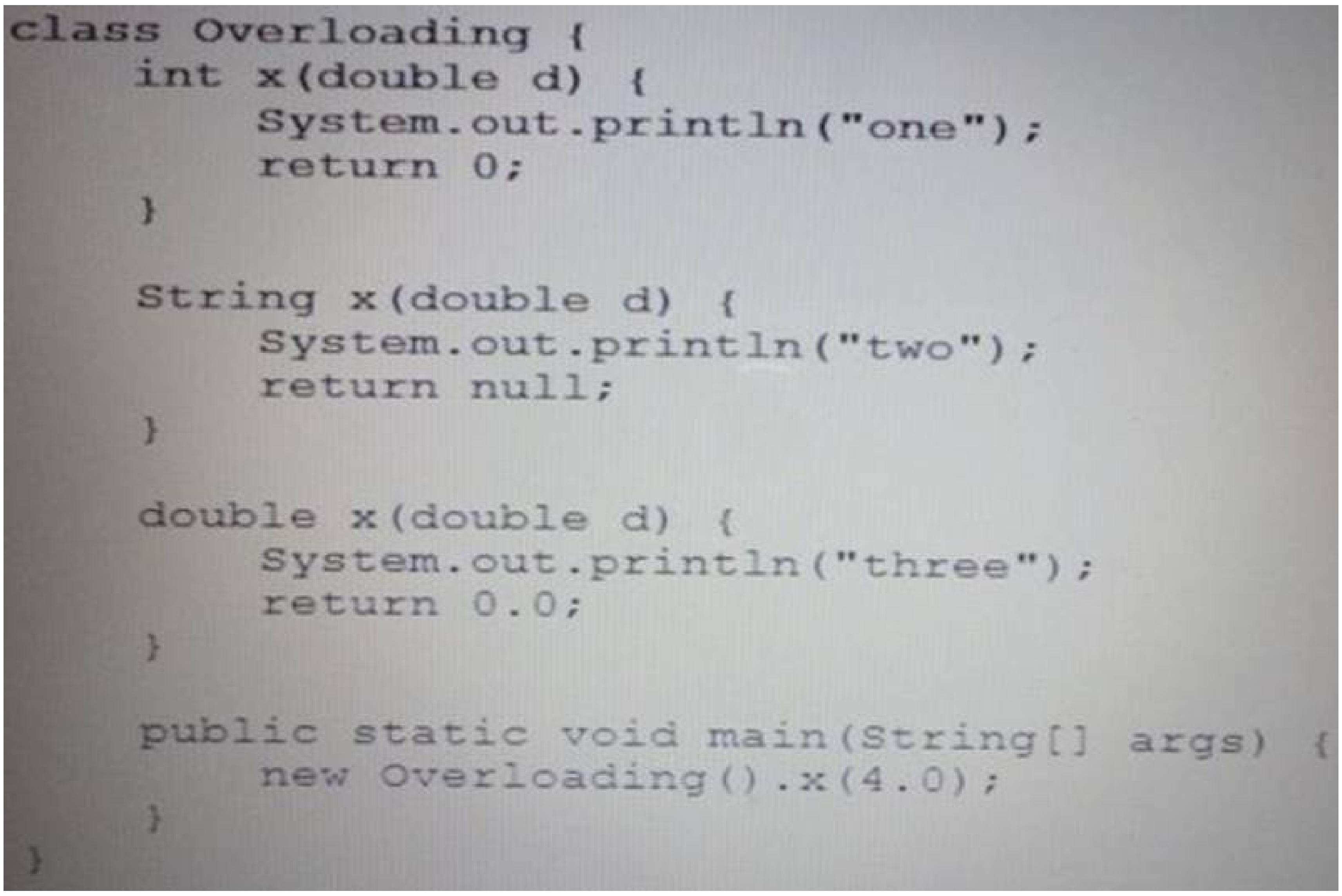
What is the result?

A. They match They really match
B. They really match
C. They match
D. Nothing Prints
E. They reallymatch They really match

Answer: B

Explanation: The strings are not the same objects so the == comparison fails. See note #1 below. As the value of the strings are the same equals is true. The equals method compares values for equality.
Note: #1 ==
Compares references, not values. The use of == with object references is generally limited to the following:
Comparing to see if a reference is null.
Comparing two enum values. This works because there is only one object for each enum constant.
You want to knowif two references are to the same object.

NEW QUESTION 346
Given:



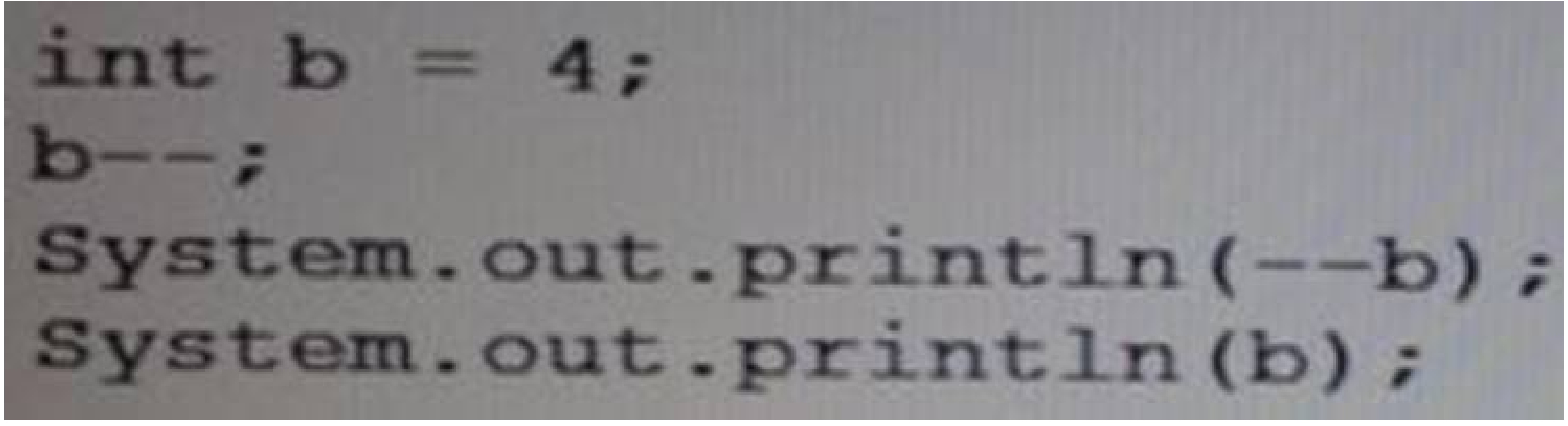
```
class Overloading {
    int x(double d) {
        System.out.println("one");
        return 0;
    }

    String x(double d) {
        System.out.println("two");
        return null;
    }

    double x(double d) {
        System.out.println("three");
        return 0.0;
    }

    public static void main(String[] args) {
        new Overloading().x(4.0);
    }
}
```

What is the result?

A. One
B. Two
C. Three
D. Compilation fails

Answer: D

NEW QUESTION 351
Given the code fragment:



```
int b = 4;
b--;
System.out.println(--b);
System.out.println(b);
```

What is the result?

A. 22
B. 12
C. 32
D. 33

Answer: A

Explanation: Variable b is set to 4.
Variableb is decreased to 3.
Variable b is decreased to 2 and then printed. Output: 2 Variable b is printed. Output: 2

NEW QUESTION 356
Given the code fragment:

```
public class Test {
public static void main(String[] args) { boolean isChecked = false;
int arry[] = {1,3,5,7,8,9};
int index = arry.length; while ( <code1> ) {
if (arry[index-1] % 2 ==0) { isChecked = true;
}
<code2>
}
System.out.print(arry(index]+", "+isChecked));
}
}
```

Which set of changes enable the code to print 1, true?

A. Replacing <code1> with index > 0 and replacing <code2> with index--;
B. Replacing <code1> with index > 0 and replacing <code2> with --index;
C. Replacing <code1> with index > 5 and replacing <code2> with --index ;
D. Replacing <code1> with index and replacing <code2> with --index ;

Answer: A

Explanation: Note: Code in B (code2 is --index;). also works fine.

NEW QUESTION 360
Given:

```
public class Equal {
public static void main(String[] args) { String str1 = "Java";
String[] str2 = {"J","a","v","a"}; String str3 = "";
for (String str : str2) { str3 = str3+str;
}
boolean b1 = (str1 == str3); boolean b2 = (str1.equals(str3)); System.out.print(b1+", "+b2);
}
```

What is the result?

A. true, false
B. false, true
C. true, true
D. false, false

Answer: B

Explanation: == strict equality. equals compare state, not identity.

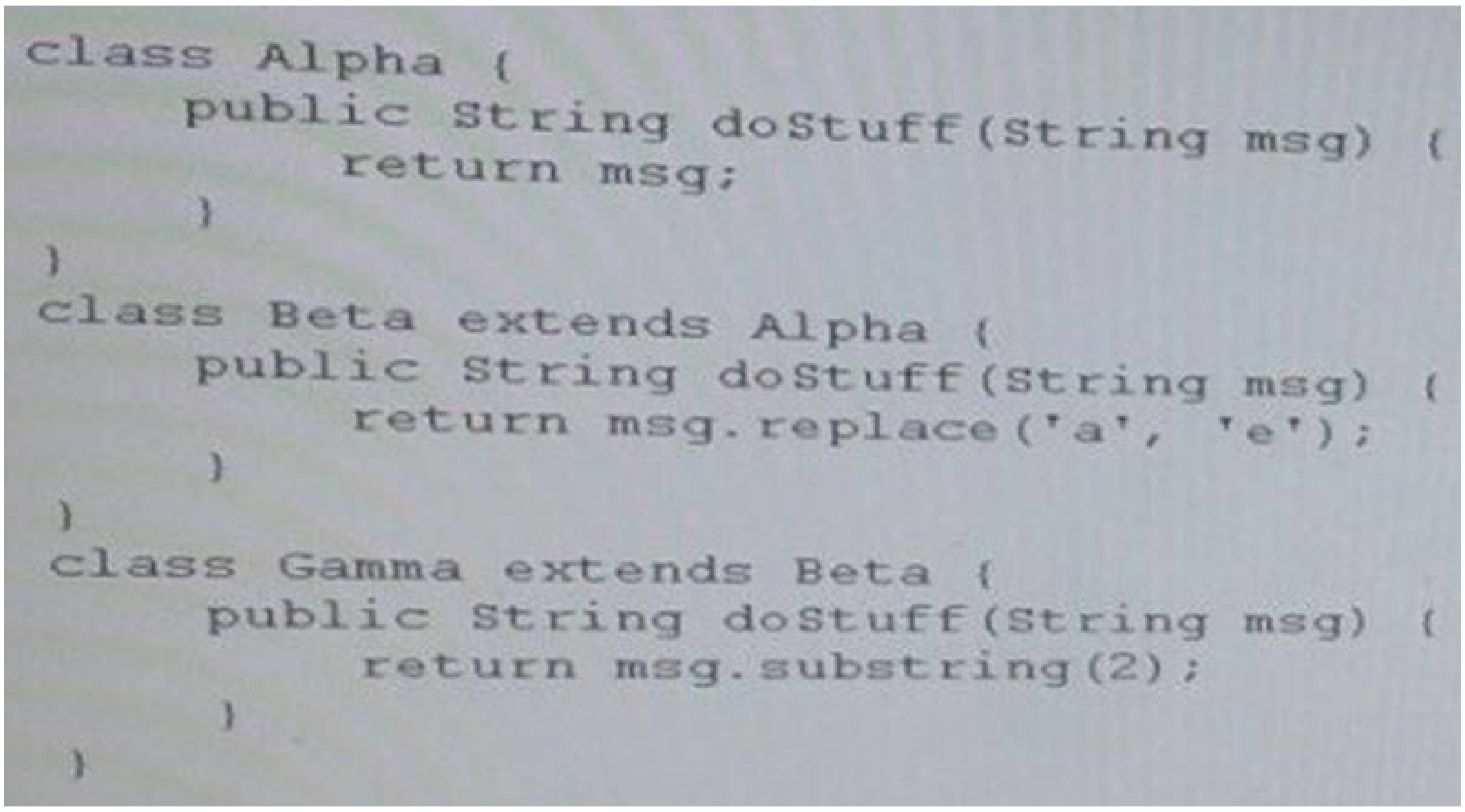
NEW QUESTION 362
Identify two benefits of using ArrayList over array in software development.

A. reduces memory footprint
B. implements the Collection API
C. is multi.thread safe
D. dynamically resizes based on the number of elements in the list

Answer: AD

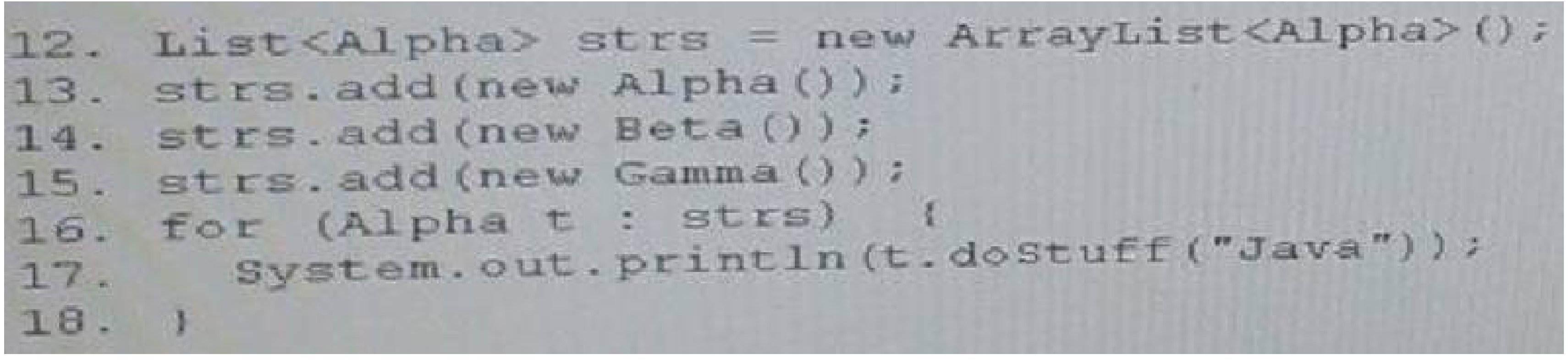
Explanation: ArrayList supports dynamic arrays that can grow as needed. In Java, standard arrays are of a fixed length. After arrays are created, they cannot grow or shrink, which means that you must know in advance how many elements an array will hold. But, sometimes, you may not know until run time precisely how large of an array you need. To handle this situation, the collections framework defines ArrayList. In essence, an ArrayList is a variable-length array ofobject references. That is, an ArrayList can dynamically increase or
decrease in size. Array lists are created with an initial size. When this size is exceeded, the collection is automatically enlarged. When objects are removed, the array may be shrunk.

NEW QUESTION 367
Given the class definitions:



```
class Alpha {
    public String doStuff(String msg) {
        return msg;
    }
}
class Beta extends Alpha {
    public String doStuff(String msg) {
        return msg.replace('a', 'e');
    }
}
class Gamma extends Beta {
    public String doStuff(String msg) {
        return msg.substring(2);
    }
}
```

And the code fragment of the main() method,



```
List<Alpha> strs = new ArrayList<Alpha>();
strs.add(new Alpha());
strs.add(new Beta());
strs.add(new Gamma());
for (Alpha t : strs) {
    System.out.println(t.doStuff("Java"));
}
```

What is the result?

A. Java Java Java
B. Java Jeve va
C. Java Jeve ve
D. Compilation fails

Answer: D

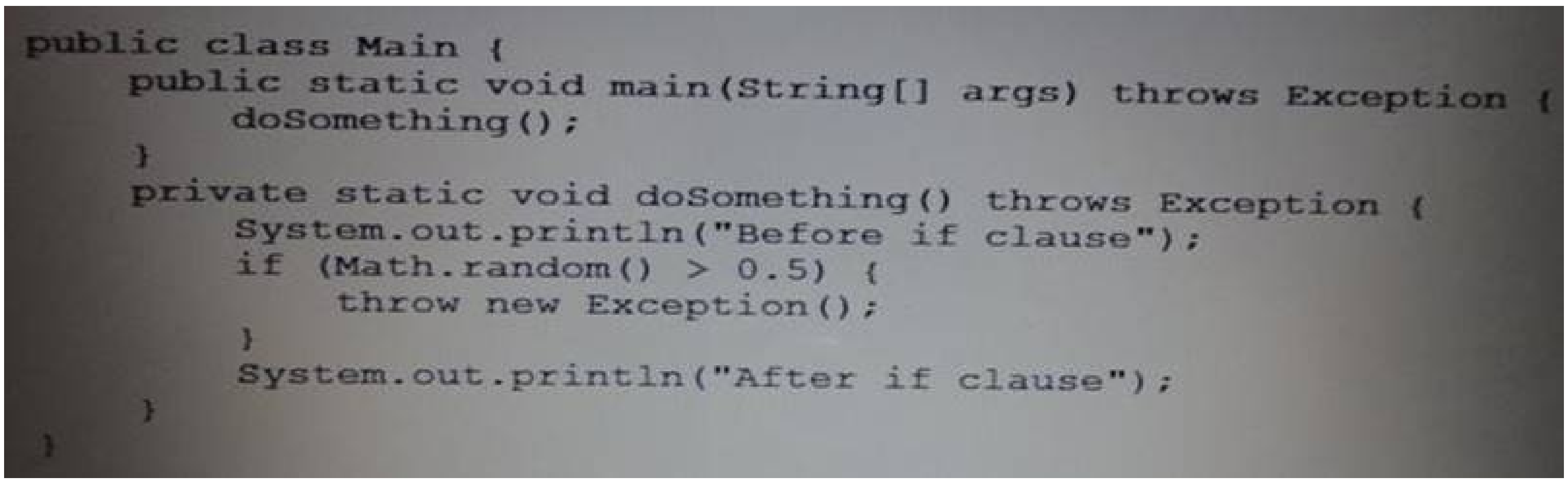
NEW QUESTION 370
Given the code fragment:
Stringh1 = "Bob";
String h2 = new String ("Bob");
What is the best way to test that the values of h1 and h2 are the same?

A. if (h1 = = h2)
B. if (h1.equals(h2))
C. if (h1 = = h2)
D. if (h1.same(h2))

Answer: B

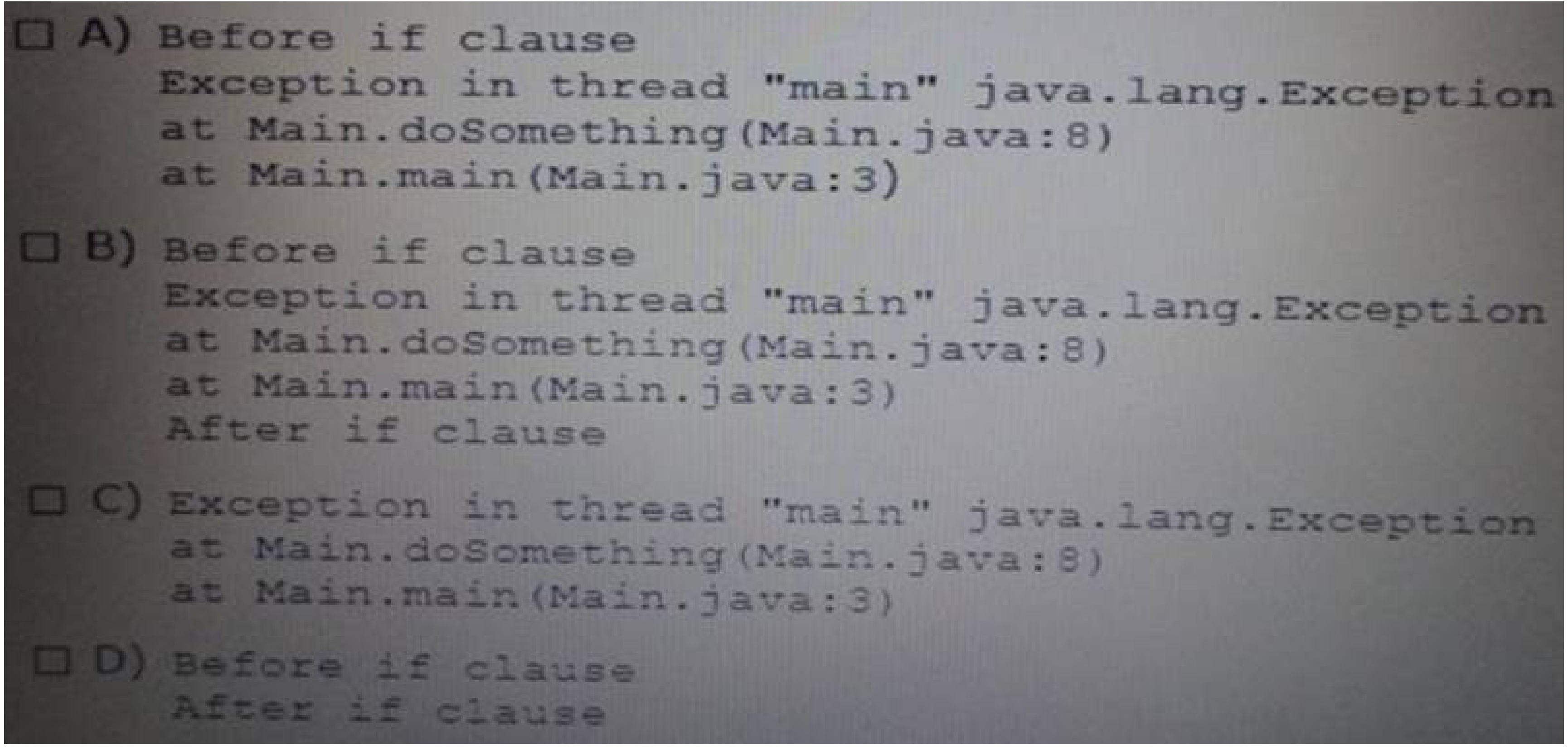
Explanation: The equals method compares values for equality.

NEW QUESTION 371
Given:



```
public class Main {
    public static void main(String[] args) throws Exception {
        doSomething();
    }
    private static void doSomething() throws Exception {
        System.out.println("Before if clause");
        if (Math.random() > 0.5) {
            throw new Exception();
        }
        System.out.println("After if clause");
    }
}
```

Which two are possible outputs?



```
□ A) Before if clause
     Exception in thread "main" java.lang.Exception
     at Main.doSomething(Main.java:8)
     at Main.main(Main.java:3)

□ B) Before if clause
     Exception in thread "main" java.lang.Exception
     at Main.doSomething(Main.java:8)
     at Main.main(Main.java:3)
     After if clause

□ C) Exception in thread "main" java.lang.Exception
     at Main.doSomething(Main.java:8)
     at Main.main(Main.java:3)

□ D) Before if clause
     After if clause
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: AD

Explanation: The first println statement, System.out.println("Before if clause");, will always run.
If Math.Random() > 0.5 then there is an exception. The exception message is displayed and the program terminates.
If Math.Random() > 0.5 is false, then the second println statement runs as well.

NEW QUESTION 374
Which statement initializes a stringBuilder to a capacity of 128?

A. StringBuilder sb = new String ("128");
B. StringBuilder sb = StringBuilder.setCapacity (128);
C. StringBuilder sb = StringBuilder.getInstance (128);
D. StringBuilder sb = new StringBuilder (128);

Answer: D

Explanation: StringBuilder(int capacity)
Constructs a string builder with no characters in it and an initial capacity specified by thecapacityargument.
Note: An instance of a StringBuilder is a mutable sequence of characters.
The principal operations on aStringBuilderare theappendandinsertmethods, which are overloaded so as to accept data of any type. Each effectively converts a given datum to a string and then appends or inserts the characters of that string to the string builder. Theappendmethod always adds these characters at the end of the builder; theinsertmethod adds the characters at a specified point.

NEW QUESTION 375
Given the code fragment: List colors = new ArrayList(); colors.add("green"); colors.add("red");
colors.add("blue"); colors.add("yellow"); colors.remove(2); colors.add(3,"cyan"); System.out.print(colors);
What is the result?

A. [green, red, yellow, cyan]
B. [green, blue, yellow, cyan]
C. [green, red, cyan, yellow]
D. Am IndexOutOfBoundsException is thrown at runtime

Answer: A

Explanation: First the list [green, red, blue, yellow] is build. The blue element is removed:
[green, red, yellow]
Finally the element cyan is added at then end of the list (index 3). [green, red, yellow, cyan]

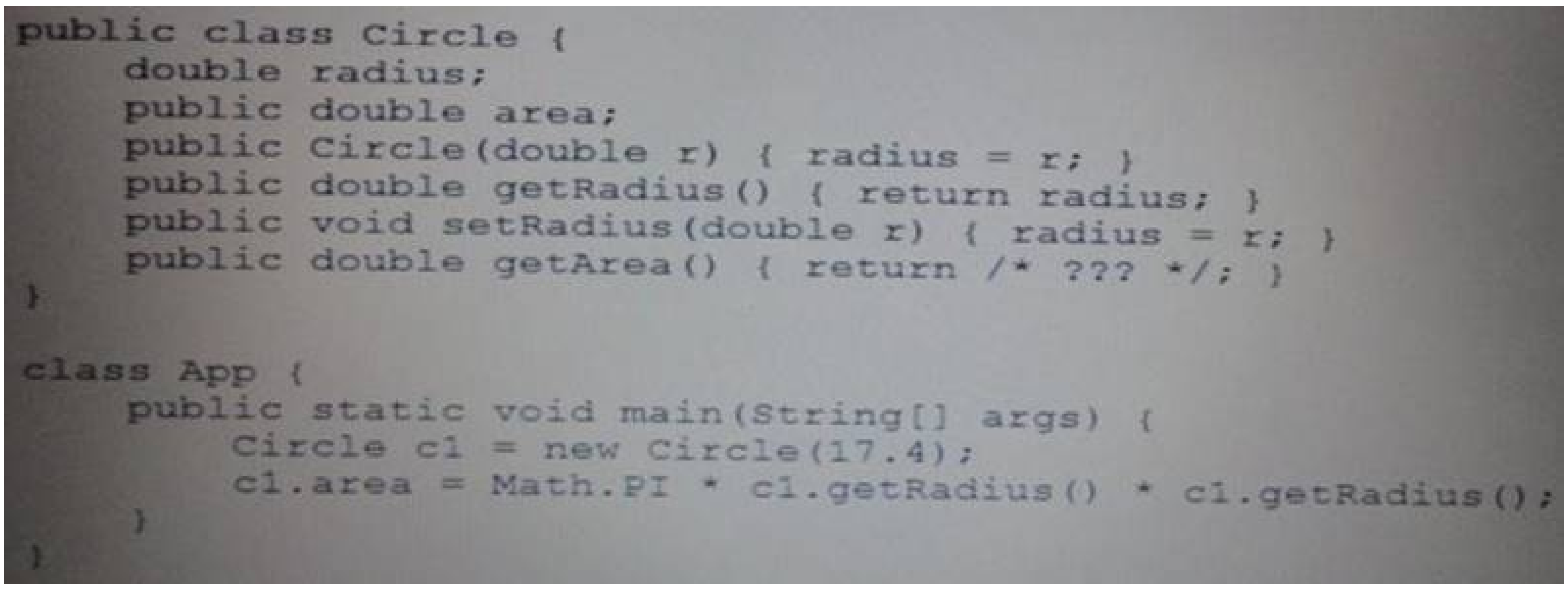
NEW QUESTION 380
View the exhibit:
public class Student {
public String name = "";
public int age = 0;
public String major = "Undeclared";

public boolean fulltime = true;
public void display() {
System.out.println("Name: " + name + " Major: " + major); }
publicboolean isFullTime() {
return fulltime;
}
}
Given:
Public class TestStudent {
public static void main(String[] args) {
Student bob = new Student ();
bob.name = "Bob";
bob.age = 18;
bob.year = 1982;
}
}
What is the result?

A. year is set to 1982.
B. bob.year is set to 1982
C. A runtime error is generated.
D. A compile time error is generated.

Answer: D

NEW QUESTION 385
Given:



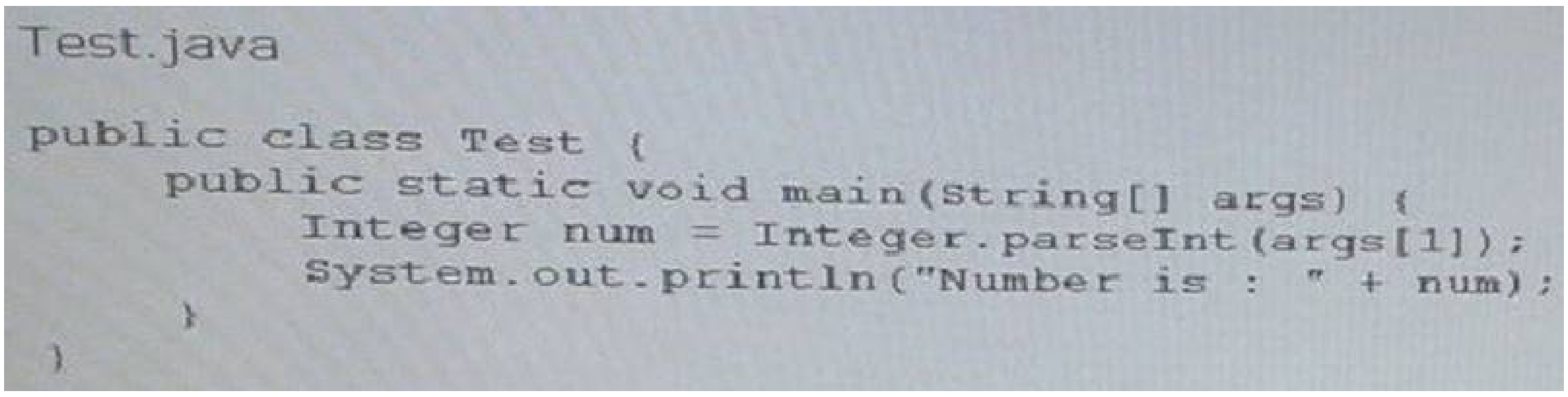
```
public class Circle {
     double radius;
     public double area;
     public Circle(double r) { radius = r; }
     public double getRadius() { return radius; }
     public void setRadius(double r) { radius = r; }
     public double getArea() { return /* ??? */; }
}

class App {
     public static void main(String[] args) {
          Circle c1 = new Circle(17.4);
          c1.area = Math.PI * c1.getRadius() * c1.getRadius();
     }
}
```

This class is poorly encapsulated. You need to change the circle class to compute and return the area instead.
What three modifications are necessary to ensurethat the class is being properly encapsulated?

A. Change the access modifier of the setradius () method to private
B. Change the getArea () method public double getArea () { return area; }
C. When the radius is set in the Circle constructor and the setRadius () method, recomputed the area and store it into the area field
D. Change the getRadius () method: public double getRadius () {area = Math.PI * radius * radius; return radius;}

Answer: BCD

NEW QUESTION 390
Given:



```
Test.java

public class Test {
     public static void main(String[] args) {
          Integer num = Integer.parseInt(args[1]);
          System.out.println("Number is : " + num);
     }
}
```

And the commands: Javac Test.java Java Test 12345
What is the result?

A. Number us : 12345
B. A NullPointerException is thrown at runtime
C. A NumberFormatException is thrown at runtime
D. AnArrayIndexOutOfBoundException is thrown at runtime.

Answer: A

NEW QUESTION 392
Given:
public class MainMethod { void main() { System.out.println("one");
}
static void main(String args) { System.out.println("two");
}
public static void main(String[] args) { System.out.println("three");
}
void mina(Object[] args) { System.out.println("four");
}
}
What is printed out when the program isexcuted?

A. one
B. two
C. three
D. four

Answer: C

NEW QUESTION 393
Given: class Mid {
public int findMid(int n1, int n2) {
return (n1 + n2) / 2;
}
}
public class Calc extends Mid {
public static void main(String[] args) {
int n1 = 22, n2 = 2;
// insert code here
System.out.print(n3);
}
}
Which two code fragments, when inserted at // insert code here, enable the code to compile and print 12?

A. Calc c = new Calc(); int n3 = c.findMid(n1,n2);
B. int n3 = super.findMid(n1,n3);
C. Calc c = new Mid();int n3 = c.findMid(n1, n2);
D. Mid m1 = new Calc();int n3 = m1.findMid(n1, n2);
E. int n3 = Calc.findMid(n1, n2);

Answer: AD

Explanation: Incorrect:
Not B: circular definition of n3.
Not C: Compilation error. line Calc c = new Mid(); required: Calc
found: Mid
Not E: Compilation error. line int n3 = Calc.findMid(n1, n2);
non-static method findMid(int,int) cannot be referenced from a static context

NEW QUESTION 398
......