QUESTION 1
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

QUESTION 2
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();
	}
```

```
}
```

What is the result?

A.
```
ns = 50 s = 125
ns = 125 s = 125
ns = 100 s = 125
```

B.
```
ns = 50 s = 125
ns = 125 s = 125
ns = 0 s = 125
```

C.
```
ns = 50 s = 50
ns = 125 s = 125
ns = 100 s = 100
```

D.
```
ns = 50 s = 50
ns = 125 s = 125
ns = 0 s = 125
```

QUESTION 3

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 { }`

QUESTION 4

Given:

```
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 >= 100) {
			throw new AgeOutOfRangeException();
		}
		else {
			this.name= name;
			this.age = age;
		}
	}
	public String toString() {
		return name + " age: " + age;
	}
}
```

And

```
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 (MissingInfoException e2) { //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}
```

QUESTION 5
Given:

```
public class Test {
      static void dispResult(int[] num) {
            try {
                  System.out.println(num[0] / (num[0] - num[1]));
            }
            catch (ArithmeticException ex) {
                  System.out.println("First exception");
            }
            System.out.println("Done");
      }
      public static void main(String[] args) {
            try {
                  int[] arr = {100, 100};
                  dispResult(arr);
            }
            catch (IllegalArgumentException ex) {
                  System.out.println("Second exception");
            }
            catch(Exception ex) {
                  System.out.println("Third exception");
            }
      }
}
```

What is the result?

A.
```
0
Done
```

B. `First exception`
`Done`
C. `Second exception`
D. `Done`
`Third exception`
E. `Third exception`

QUESTION 6
Given:

```
class X {
      public void mX() {
            System.out.println("Xm1");
      }
}
class Y extends X {
      public void mX() {
            System.out.println("Xm2");
      }
      public void mY() {
            System.out.println("Ym");
      }
}

public class Test {
      public static void main(String[] args) {
            X xRef = new Y();
            Y yRef = (Y)xRef;
            yRef.mY();
            xRef.mX();
      }
}
```

What is the result?
A. `Ym`
`Xm2`
B. `Ym`
`Xm1`
C. `Compilation fails`
D. `A ClassCastException is thrown at runtime`

QUESTION 7
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 utc = Character.toString(message.charAt
                                    (pos)).toUpperCase();
                                    message.replace(pos, pos+1, utc);
                              }
                        }
                  } catch (Exception ex) {
                        System.out.println("Out of limits");
                  }
                  System.out.println(message);
      }
}
```

What is the result?

A. `hEllO jAvA!`
B. `Hello java!`
C. `Out of limits`
`hEllO jAvA!`
D. `Out of límits`

QUESTION 8
Given:

```
3. public class OffRamp {
      4. public static void main(String[] args) {
            5. int [] exits = {0,0,0,0,0,0};
            6. int x1 = 0;
            7.
            8. for(int x = 0; x < 4; x++) exits[0] = x;
            9. for(int x = 0; x < 4; ++x) exits[1] = x;
            10.
            11. x1 = 0; while(x1++ < 3) exits[2] = x1;
            12. x1 = 0; while(++x1 < 3) exits[3] = x1;
            13.
            14. x1 = 0; do { exits[4] = x1; } while(x1++ < 7);
            15. x1 = 0; do { exits[5] = x1; } while(++x1 < 7);
            16.
            17. for(int x: exits)
                  18. System.out.print(x + " ");
19. } }
```

What is the result?

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

QUESTION 9
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`

QUESTION 10
Given:

```
package p1;
public interface DoInterface {
	void m1(int n); // line n1
	public void m2(int n);
}
package p3;
import p1.DoInterface;
public class DoClass implements DoInterface {
	int x1, x2;
	DoClass() {
		this.x1 = 0;
		this.x2 = 10;
	}
	public void m1(int p1) {x1 += p1; System.out.println(x1);} //line n2
	public void m2(int p1) {x2 += p1; System.out.println(x2);}
}
package p2;
import p1.*;
import p3.*;
class Test {
	public static void main(String[] args) {
		DoInterface doi = new DoClass(); // line n3
		doi.m1(100);
		doi.m2(200);
	}
}
```

What is the result?

A. 100
210
B. Compilation fails due to an error in line n1
C. Compilation fails due to an error at line n2
D. Compilation fails due to an error at line n3

QUESTION 11
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

QUESTION 12
Given the code fragment:

```
public class App {
	public static void main(String[] args) {
		int[] lst = {1, 2, 3, 4, 5, 4, 3, 2, 1};
		int sum = 0;
		for(int frnt = 0, rear = lst.length - 1; frnt < 5 && rear >= 5;
		frnt++, rear--) {
			sum = sum + lst[frnt] + lst[rear];
		}
		System.out.println(sum);
	}
}
```

What is the result?

A. 20
B. 25
C. 29
D. Compilation fails
E. AnArrayIndexOutOfBoundsException is thrown at runtime

QUESTION 13
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. java.lang.StringIndexOutOfBoundsException is thrown at runtime.
D. java.lang.ArrayIndexOutOfBoundsException is thrown at runtime.
E. java.lang.NullPointerException is thrown at runtime.

QUESTION 14
Given:

```
public class MyFor3 {
	public static void main(String[] args) {
		int[] xx = null;
```

```
            for (int i : xx) {
                  System.out.println(i);
            }
      }
}
```

What is the result?

A. null
B. Compilation fails
C. An exception is thrown at runtime
D. 0

QUESTION 15

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 ex) {
                              System.out.println("String out of limits");
                        }
                  }
            }catch(ArrayIndexOutOfBoundsException ex) {
                  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

QUESTION 16

Given:

```
public class Case {
      public static void main(String[] args) {
            String product = "Pen";
```

```
            product.toLowerCase();
            product.concat(" BOX".toLowerCase());
            System.out.println(product.substring(4, 6));
      }
}
```

What is the result?

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

QUESTION 17

Given the code fragment:

```
public class Demo {
      public static void main(String[] args) {
            int aVar = 9;
            if (aVar++ < 10) {
                  System.out.println(aVar + " Hello world!");
            }
            else {
                  System.out.println(aVar + " Hello universe!");
            }
      }
}
```

What is the result?

A. 10 Hello world!
B. 10 Hello universe!
C. 9 Hello world!
D. Compilation fails.

QUESTION 18

Given:

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

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

QUESTION 19
Given:

```
public class TestApp {
      public static void main(String[] args) {
            TestApp t = new TestApp();
            try {
                  t.doPrint();
                  t.doList();
            }
            catch(Exception ex) {
                  System.out.println("Caught " + ex);
            }
      }
      public void doList() throws Exception {
            throw new Error("Error");
      }
      public void doPrint() throws Exception {
            throw new RuntimeException("Exception");
      }
}
```

What is the result?

A.
```
Caught java.lang.RuntimeException: Exception
Exception in thread "main" java.lang.Error: Error
at TestApp.doList(TestApp.java: 14)
at TestApp.main(TestApp.java: 6)
```
B.
```
Exception in thread "main" java.lang.Error: Error
at TestApp.doList(TestApp.java: 14)
at TestApp.main(TestApp.java: 6)
```
C.
```
Caught java.lang.RuntimeException: Exception
Caught java.lang.Error: Error
```
D.
```
Caught java.lang.RuntimeException: Exception
```

QUESTION 20
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. Change the getArea( ) method as follows:

```
public double getArea ( ) {area = Match.PI * radius * radius; }
```

D. Change the access modifier of the setRadius ( ) method to be protected.