

OCP QUESTION 2

Given the class definitions:

```
class Plant {  
    public String brew(String msg) {  
        return msg;  
    }  
}  
class Coffee extends Plant {  
    public String brew(String msg) {  
        return msg.replace('a', 'e');  
    }  
}  
class Arabica extends Coffee {  
    public String brew(String msg) {  
        return msg.substring(2);  
    }  
}
```

And the code fragment of the main() method,

```
List<Plant> cup = new ArrayList<Plant>();  
cup.add(new Plant());  
cup.add(new Coffee());  
cup.add(new Arabica());  
for (Plant item : cup) {  
    System.out.println(item.brew("Java"));  
}
```

What is the result?

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

OCP QUESTION 13

Given:

```
public final class WordPress {
    public void login() {}
}

public class Post {
    public final void write(int words, int pix) {}
    public void uploadMedia() {}
}

public class FrontPage {
    private Post p = new Post();
    private final String subj = "OCA & OCP Prep Tools";
    public void compose(){ p.write(1200, 2); }
}

public class StickyPost extends Post {
    public void write(int numberOfWords, int numberOfPix) {}
    public void publish() {}
}
```

Which statement is true?

- A. A compilation error occurs in **WordPress**.
- B. A compilation error occurs in **Post**.
- C. A compilation error occurs in **FrontPage**.
- D. A compilation error occurs in **StickyPost**.
- E. All classes compile successfully.

OCP QUESTION 16

Given the code fragment:

```
class Hollywood{  
    public static void main(String[] args) {  
        String title = "MUMMY STRIKES BACK!";  
        System.out.println("Title = " + title.replace("M", "P"));  
    }  
}
```

What is the result?

- A. Title = MUMMY STRIKES BACK!
- B. Title = PUPPY STRIKES BACK!
- C. A compile time error is produced.
- D. A runtime error is produced.
- E. Title =
- F. Title = PUMMY STRIKES BACK!

OCP QUESTION 20

Given:

```
class WhiteStarLiner implements Sinkable {
    public void sink() { }
}

abstract class Shipwreck extends WhiteStarLiner { }

class Britannic extends WhiteStarLiner {
    protected void sink(int numberOfMines) { }
}

class Titanic extends WhiteStarLiner implements Sinkable {
    public void hitIceberg() { }
}

interface Sinkable {
    public abstract void sink();
}
```

Which statement is true?

- A. Shipwreck does not compile.
- B. Britannic does not compile.
- C. Titanic does not compile.
- D. Sinkable does not compile.
- E. All classes compile successfully.

OCP QUESTION 34

Given:

Bread.java:

```
public class Bread {  
    private String eat(String piece) { return "Consume " + piece; }  
}
```

Pizza.java:

```
public class Pizza extends Bread {  
    public String eat (String slice) { return "Enjoy " + slice; }  
}
```

Test.java:

```
public class Test {  
    public static void main (String[] args) {  
        Bread b1 = new Bread();  
        b1.eat("bread.");  
        Bread b2 = new Pizza();  
        b2.eat("Quattro Stagioni.");  
    }  
}
```

What is the result?

- A. Consume bread.
Enjoy Quattro Stagioni.
- B. Consume bread.
Consume Quattro Stagioni.
- C. The Pizza.java file fails to compile.
- D. The Test.java file fails to compile.

OCP QUESTION 55

Given the code:

```
class AccessKey{
    String user;
    String pass;
}
```

and

```
12. class DBAccess{
13.     AccessKey grantAccess(String user, String pass){
14.         // insert code here
15.     }
16.     public static void main(String[] args){
17.         DBAccess dba = new DBAccess();
18.         AccessKey key = dba.grantAccess("Bill", "1234");
19.     }
20. }
```

Which code fragment must be inserted at line 14 to enable the code to compile?

- A. AccessKey x; return x;
- B. return AccessKey;
- C. return new AccessKey();
- D. return 0;

OCP QUESTION 58

Given:

```
public class B implements A {
    public String toString() {
        return "B ";
    }
    public static void main(String[] args) {
        C myC = new C();
        B myB = myC;
        A myA = myB;
        System.out.print(myB);
        System.out.print((C) myB);
        System.out.print(myA);
    }
}
class C extends B {
    public String toString() {
        return "C ";
    }
}
interface A {
    public String toString();
}
```

What is the result?

- A. B B B
- B. B C B
- C. C C B
- D. C C C
- E. The code throws a ClassCastException

OCP QUESTION 60

Given:

```
class A {
    public void runA() {
        System.out.println("ArunA");
    }
}
class B extends A {
    public void runA() {
        System.out.println("BrunA");
    }
    public void runB() {
        System.out.println("BrunB");
    }
}
public class Test {
    public static void main(String[] args) {
        A a = new B();
        B b = (B)a;
        b.runB();
        a.runA();
    }
}
```

What is the result?

- A. BrunB
BrunA
- B. BrunB
ArunA
- C. Compilation fails
- D. A ClassCastException is thrown at runtime

OCP QUESTION 62

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. square
circle
...
- D. Compilation fails

OCP QUESTION 63

Given the code fragments:

```
interface Surveillable {}
class Target implements Surveillable {}
class PrimeSuspect extends Target {}
class Mission {
    public static void main(String[] args) {
        List objectives = new ArrayList();
        Surveillable s1 = new Target();
        Surveillable s2 = new PrimeSuspect();           // line n1
        Target t1 = new PrimeSuspect();
        objectives.add(s1);
        objectives.add(s2);
        objectives.add(t1);                             // line n2
        for (Object item : objectives) {
            System.out.println(item.getClass().getName());
        }
    }
}
```

What is the result?

- A. Target
PrimeSuspect
PrimeSuspect
- B. Surveillable
Surveillable
Target
- C. Compilation fails at line n1
- D. Compilation fails at line n2

OCP QUESTION 67

Given the code fragment:

```
public class Test {  
    public static void main(String[] args) {  
        int[] array = {1,2,3};  
        for ( missing_code ) {  
            }  
        }  
    }  
}
```

Which three are valid replacements for *missing_code* so that the program will compile and run?

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

OCP QUESTION 68

Given the definition of the following class:

```
class LongDistanceVoIP {
    double tariff;
    LongDistanceVoIP(double tariff) {
        this.tariff = tariff;
    }
    public void makeCall(int time) {
        int minutes = time;
        class CallingChicago {                                // line n1
            double charge = 0;
            public void calcCost() {
                charge = minutes * tariff;                    // line n2
                System.out.println("It'll set me back for " + charge + " cents.");
            }
        }
        new CallingChicago().calcCost();                      // line n3
    }
}
```

and this code fragment:

```
LongDistanceVoIP ld = new LongDistanceVoIP(1.1);
ld.makeCall(10);
```

What is the result?

- A. It'll set me back for 11.0 cents.
- B. A compilation error occurs at line n1.
- C. A compilation error occurs at line n2.
- D. A compilation error occurs at line n3.

OCP QUESTION 69

Given:

```
class Puzzler {
    int tally = 0;
    public void doStuff(int val) {
        if (val % 2 == 0) {
            break;
        } else {
            for (int i = 0; i < val; i++) {
                tally += i;
            }
        }
    }
    public static void main(String[] args) {
        Puzzler obj = new Puzzler();
        System.out.println("Left " + obj.tally);
        obj.doStuff(4);
        System.out.println("Middle " + obj.tally);
        obj.doStuff(5);
        System.out.println("Right " + obj.tally);
    }
}
```

What is the result?

- | | | | | |
|---|--|------------------------------|-------------------|-----------------------------|
| A. Left 0
Middle 0
Right 5 | B. Left 0
Middle 0
Right 10 | C. Left 0
Middle 3 | D. Right 6 | E. Compilation fails |
|---|--|------------------------------|-------------------|-----------------------------|

OCP QUESTION 79

Given:

```
public class Test{  
    public static void main(String[] args) {  
        int x = 10;  
        int y = 20;  
        int z = y += x/5;  
        System.out.print(x + " : " + y + " : " + z);  
    }  
}
```

What is the result?

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

OCP QUESTION 92

Given:

```
public class LetterFromJane {
    public static void main(String[] args) {
        StringBuilder letter = new StringBuilder("Dear John");
        int indx = 0;
        try {
            for (indx = 0; indx < 10; indx++) {
                switch (letter.charAt(indx)) {
                    case 'a':
                    case 'e':
                    case 'o':
                        String uc = Character.toString(letter.charAt(indx))
                                                .toUpperCase();
                        letter.replace(indx, indx + 1, uc);
                }
            }
        } catch (Exception e) {
            System.out.println("Goodbye.");
        }
        System.out.println(letter);
    }
}
```

What is the result?

- A. DEAr JOhn
- B. Dear John
- C. Goodbye.
DEAr JOhn
- D. Goodbye.

OCP QUESTION 96

Given:

ExamTaker.java:

```
public class ExamTaker {
    private String fName;
    private String lName;
    private static int count;
    public ExamTaker(String first, String last) {
        fName = first;
        lName = last;
        ++count;
    }
    static {
        count = 0;
    }
    public static int getCount() {
        return count;
    }
}
```

Test.java:

```
public class Test {
    public static void main(String[] args) {
        ExamTaker et1 = new ExamTaker("Alice", "Adams");
        ExamTaker et2 = new ExamTaker("Bob", "Barry");
        ExamTaker et3 = new ExamTaker("Chuck", "Collins");
        ExamTaker et4 = new ExamTaker("Doug", "Dowson");
        et4 = null;
        et3 = et2;
        System.out.println(ExamTaker.getCount());
    }
}
```

What is the result?

A. 0

B. 2

C. 3

D. 4

E. 5