CSE143 Section #11 Problems

1. Assuming that the following classes have been defined:

```
public class Foo {
    public void method1() {
        System.out.println("foo 1");
    }
}

public class Bar extends Foo {
    public void method1() {
        System.out.println("bar 1");
        super.method1();
        method2();
    }

    public void method2() {
        System.out.println("bar 2");
    }
}

public class Baz extends Bar {
    public void method2() {
        System.out.println("baz 2");
    }
}
```

And assuming the following variables have been defined:

```
Bar var1 = new Baz();
Object var2 = new Bar();
```

Indicate the output produced (or error caused) by each statement below.

```
Statement                          Output
-----------------------------------------------------

var1.method1();                    _____

((Baz)var2).method2();             _____

((Bar)var2).method2();             _____
```

2. Assuming that the following classes have been defined:

```java
public class First {
    public void method2() {
        System.out.println("First2");
    }

    public void method3() {
        method2();
    }
}

public class Second extends First {
    public void method2() {
        System.out.println("Second2");
    }
}

public class Third extends Second {
    public void method1() {
        System.out.println("Third1");
        super.method2();
    }

    public void method2() {
        System.out.println("Third2");
    }
}

public class Fourth extends First {
    public void method1() {
        System.out.println("Fourth1");
    }

    public void method2() {
        System.out.println("Fourth2");
    }
}
```

And assuming the following variables have been defined:

```
First var1 = new Second();
First var2 = new Third();
First var3 = new Fourth();
Second var4 = new Third();
Object var5 = new Fourth();
Object var6 = new Second();
```

In the table below, indicate in the right-hand column the output produced by
the statement in the left-hand column. If the statement produces more than
one line of output, indicate the line breaks with slashes as in "a/b/c" to
indicate three lines of output with "a" followed by "b" followed by "c". If
the statement causes an error, fill in the right-hand column with either the
phrase "compiler error" or "runtime error" to indicate when the error would
be detected.

```
Statement                        Output
-----------------------------------------------------------

var1.method2();                  _____

var2.method2();                  _____

var3.method2();                  _____

var4.method2();                  _____

var5.method2();                  _____

var6.method2();                  _____

var1.method3();                  _____

var2.method3();                  _____

var3.method3();                  _____

var4.method3();                  _____

var5.method3();                  _____

var6.method3();                  _____

((Second)var4).method1();        _____

((Third)var4).method1();         _____

((Second)var5).method2();        _____

((First)var5).method3();         _____

((Third)var5).method1();         _____

((First)var6).method3();         _____

((Second)var6).method1();        _____

((Second)var6).method3();        _____
```

3. Assuming that the following classes have been defined:

```java
public class Harry {
    public void method1() {
        System.out.println("harry 1");
    }

    public void method2() {
        method1();
        System.out.println("harry 2");
    }
}

public class Larry extends Harry {
    public void method1() {
        System.out.println("larry 1");
        super.method1();
    }
}

public class Mary extends Larry {
    public void method2() {
        System.out.println("mary 2");
    }

    public void method3() {
        super.method1();
        System.out.println("mary 3");
    }
}

public class Jerry extends Mary {
    public void method2() {
        super.method2();
        System.out.println("jerry 2");
    }
}
```

And assuming the following variables have been defined:

```
Harry var1 = new Harry();
Harry var2 = new Larry();
Larry var3 = new Jerry();
Mary var4 = new Mary();
Mary var5 = new Jerry();
Object var6 = new Larry();
```

In the table below, indicate in the right-hand column the output produced by
the statement in the left-hand column. If the statement produces more than
one line of output, indicate the line breaks with slashes as in "a/b/c" to
indicate three lines of output with "a" followed by "b" followed by "c". If
the statement causes an error, fill in the right-hand column with either the
phrase "compiler error" or "runtime error" to indicate when the error would
be detected.

```
Statement                         Output
------------------------------------------------------------

var1.method1();                   _____

var2.method1();                   _____

var3.method1();                   _____

var4.method1();                   _____

var5.method1();                   _____

var6.method1();                   _____

var1.method2();                   _____

var2.method2();                   _____

var3.method2();                   _____

var4.method2();                   _____

var5.method2();                   _____

var6.method2();                   _____

var3.method3();                   _____

var5.method3();                   _____

((Larry)var1).method1();          _____

((Mary)var2).method2();           _____

((Jerry)var5).method1();          _____

((Mary)var3).method3();           _____

((Jerry)var4).method3();          _____

((Mary)var6).method3();           _____
```

4. Assuming that the following classes have been defined:

```java
public class Foo {
    public void method1() {
        System.out.println("foo 1");
        method2();
    }

    public void method2() {
        System.out.println("foo 2");
    }
}

public class Bar extends Foo {
    public void method2() {
        System.out.println("bar 2");
    }

    public void method3() {
        System.out.println("bar 3");
    }
}

public class Baz extends Foo {
    public void method1() {
        System.out.println("baz 1");
    }

    public void method2() {
        System.out.println("baz 2");
        method1();
    }
}

public class Mumble extends Baz {
    public void method1() {
        super.method1();
        System.out.println("mumble 1");
    }

    public void method3() {
        System.out.println("mumble 3");
    }
}
```

And assuming the following variables have been defined:

```
Foo var1 = new Bar();
Foo var2 = new Mumble();
Bar var3 = new Bar();
Baz var4 = new Baz();
Baz var5 = new Mumble();
Object var6 = new Baz();
```

In the table below, indicate in the right-hand column the output produced by
the statement in the left-hand column. If the statement produces more than
one line of output, indicate the line breaks with slashes as in "a/b/c" to
indicate three lines of output with "a" followed by "b" followed by "c". If
the statement causes an error, fill in the right-hand column with either the
phrase "compiler error" or "runtime error" to indicate when the error would
be detected.

```
    Statement                       Output
    ------------------------------------------------------------

    var1.method1();                 _____

    var2.method1();                 _____

    var3.method1();                 _____

    var4.method1();                 _____

    var5.method1();                 _____

    var6.method1();                 _____

    var1.method2();                 _____

    var2.method2();                 _____

    var3.method2();                 _____

    var4.method2();                 _____

    var5.method2();                 _____

    var6.method2();                 _____

    var3.method3();                 _____

    var5.method3();                 _____

    ((Bar)var1).method3();          _____

    ((Mumble)var4).method3();       _____

    ((Mumble)var5).method3();       _____

    ((Bar)var2).method3();          _____

    ((Baz)var2).method2();          _____

    ((Mumble)var6).method2();       _____
```

5. Assuming that the following classes have been defined:

```
public class Fee extends Fo {
    public void method1() {
        System.out.println("Fee 1");
        super.method3();
    }

    public void method3() {
        System.out.println("Fee 3");
    }
}

public class Fie extends Fum {
    public void method1() {
        System.out.println("Fie 1");
    }

    public void method3() {
        System.out.println("Fie 3");
        super.method3();
    }
}

public class Fo {
    public void method2() {
        System.out.println("Fo 2");
        method3();
    }

    public void method3() {
        System.out.println("Fo 3");
    }
}

public class Fum extends Fo {
    public void method3() {
        System.out.println("Fum 3");
    }
}
```

And assuming the following variables have been defined:

```
Fum var1 = new Fie();
Object var2 = new Fum();
Fo var3 = new Fee();
Object var4 = new Fie();
Object var5 = new Fo();
Fum var6 = new Fum();
```

In the table below, indicate in the right-hand column the output produced by
the statement in the left-hand column. If the statement produces more than
one line of output, indicate the line breaks with slashes as in "a/b/c" to
indicate three lines of output with "a" followed by "b" followed by "c". If
the statement causes an error, fill in the right-hand column with either the
phrase "compiler error" or "runtime error" to indicate when the error would
be detected.

```
    Statement                       Output
    -----------------------------------------------------------

    var1.method2();                 _____

    var2.method2();                 _____

    var3.method2();                 _____

    var4.method2();                 _____

    var5.method2();                 _____

    var6.method2();                 _____

    var1.method3();                 _____

    var2.method3();                 _____

    var3.method3();                 _____

    var4.method3();                 _____

    var5.method3();                 _____

    var6.method3();                 _____

    ((Fee)var3).method1();          _____

    ((Fee)var4).method1();          _____

    ((Fie)var1).method2();          _____

    ((Fo)var3).method3();           _____

    ((Fie)var6).method2();          _____

    ((Fo)var2).method3();           _____

    ((Fie)var3).method1();          _____

    ((Fo)var5).method2();           _____
```