

Exercise 3

1. This question was asked in a previous exam!

Subtyping:

- What is behavioral subtyping?
- Describe the three main rules of behavioral subtyping.
- Write a small Java program that illustrates one of the problems that might occur, if two classes are subtypes, but not behavioral subtypes.

2. Improve the following code:

```
@Override
public boolean equals(Object obj) {
    if (obj==null || !(obj instanceof StaticCallVariable)) {
        return false;
    } else {
        if (callingMethod.equals(
            ((StaticCallVariable) obj).getCallingMethod())
            && calledMethod.equals(
            ((StaticCallVariable) obj).getCalledMethod())
            && instrLocation ==
            ((StaticCallVariable) obj).getInstrumentedLocation()) {
            return true;
        } else {
            return false;
        }
    }
}
```

3. Overriding and Overloading example. Consider the following classes:

```
class Upper {}
class Middle extends Upper {}
class Lower extends Middle {}

class Super {
    Middle foo( Middle a1 ) {
        System.out.println(" Super.foo(" + a1 + ")");
        return a1;
    }

    Middle bar( Middle a1 ) {
        System.out.println(" Super.bar(" + a1 + ")");
        return a1;
    }
}

class Sub extends Super {
    Lower foo( Upper a1 ) {
        System.out.println(" Sub.foo(" + a1 + ")");
        return new Lower();
    }

    Middle bar( Middle a1 ) {
        System.out.println(" Sub.bar(" + a1 + ")");
        return a1;
    }
}
```

```
}
```

and the following main method:

```
public static void main( String[] args ) {
    Super super1;
    Sub    sub1;

    Lower lower1 = new Lower();
    Upper upper1 = new Upper();

    System.out.println("Calls on Super object:");

    super1 = new Super();
    super1.foo( lower1 );
    super1.bar( lower1 );

    super1.foo( upper1 );
    super1.bar( upper1 );

    System.out.println("\nCalls on Sub object in Super
reference:");

    super1 = new Sub();

    super1.foo( lower1 );
    super1.bar( lower1 );

    super1.foo( upper1 );
    super1.bar( upper1 );

    System.out.println("\nCalls on Sub object in Sub reference:");

    sub1 = new Sub();

    sub1.foo( lower1 );
    sub1.bar( lower1 );

    sub1.foo( upper1 );
    sub1.bar( upper1 );
}
```

Is the above program valid Java? Which methods in Sub are overwritten and which are overloaded? What output is produced by valid statements?