Principles of Software Quiz 4
Mar. 5, 2020
10 points total

Question 1. (6 pts) Consider the code below.

```java
import java.util.List;
import java.util.ArrayList;
public class Foo {
    private Bar bar;
    public Foo(String s) { this(new Bar(s)); }
    public Foo(Bar b) { this.bar = new Bar(b.getVal().toLowerCase()); }
    public Foo(Foo another) { this.bar = another.bar; }
    public void print() { bar.print(); }
    public int count() { return bar.getVal().length(); }
    public Foo getSelf() { return this; }
    public Foo getCopy() { return new Foo(this); }
    public List<Bar> getAsList() {
      List<Bar> res = new ArrayList<Bar>();
      res.add(bar);
      return res;
    }
}
class Bar {
    private String s;
    public Bar(String s) { this.s = s; }
    public Bar(Bar another) { this.s = another.getVal(); }
    public String getVal() { return this.s; }
    public void setVal(String s) { this.s = s; }
    public void print() { System.out.println(s); }
}
```

There might be one or several instances of representation exposure here. Write a minimal code snippet that reveals the most dangerous rep exposure. Explain why it is dangerous. If there is no representation exposure, write "None".

Question 2. (2 pts) If *every* public method in an ADT implementation preserves the representation invariant, and *every* public constructor guarantees the representation invariant at exit, then a client never sees an object (i.e., an instance of the ADT implementation) in an invalid (inconsistent) state.

(a) true
(b) false

Question 3. (2 pts) White-box testing ignores the code of the program.

(a) true
(b) false