Final Exam

COSC 4313 3.0 Software Engineering Testing

Section M, Winter 2005

Family Name: _____

Given Name(s): _____

Student Number: |___ |___ |___ |___ |___ |___ |___ |___ |___ |

Question	Out of	Mark
Q1	8	
Q2	10	
Q3	8	
Q4	18	
Q5	12	
Q6	22	
Q7	22	
Total	100	
Letter grade		

1. **[8 marks]** In this course, we saw many instances of the testing code "mirroring" the implementation under test. Identify two such instances, and present a brief example for each one.

   ANSWER:

2. **[10 marks]** Consider the following four classes.

```
public class Rectangle {

  public Rectangle(int h,int w){
    height = h;
    width = w;
  }

  protected int height;
  protected int width;

  public void setheight(int h){
    height = h;
  }

  public void setwidth(int w){
    width = w;
  }

  public int area (){
    return height * width;
  }
}
```

```
public class Square
  extends Rectangle {

  public Square(int side) {
    super(side, side);
  }

  public void setheight(int h) {
    height = h;
    width = h;
  }

  public void setwidth(int w) {
    height = w;
    width = w;
  }
}
```

```
public class SquareTest
  extends RectangleTest {

  protected Rectangle newRect(){
    return new Square(1);
  }
}
```

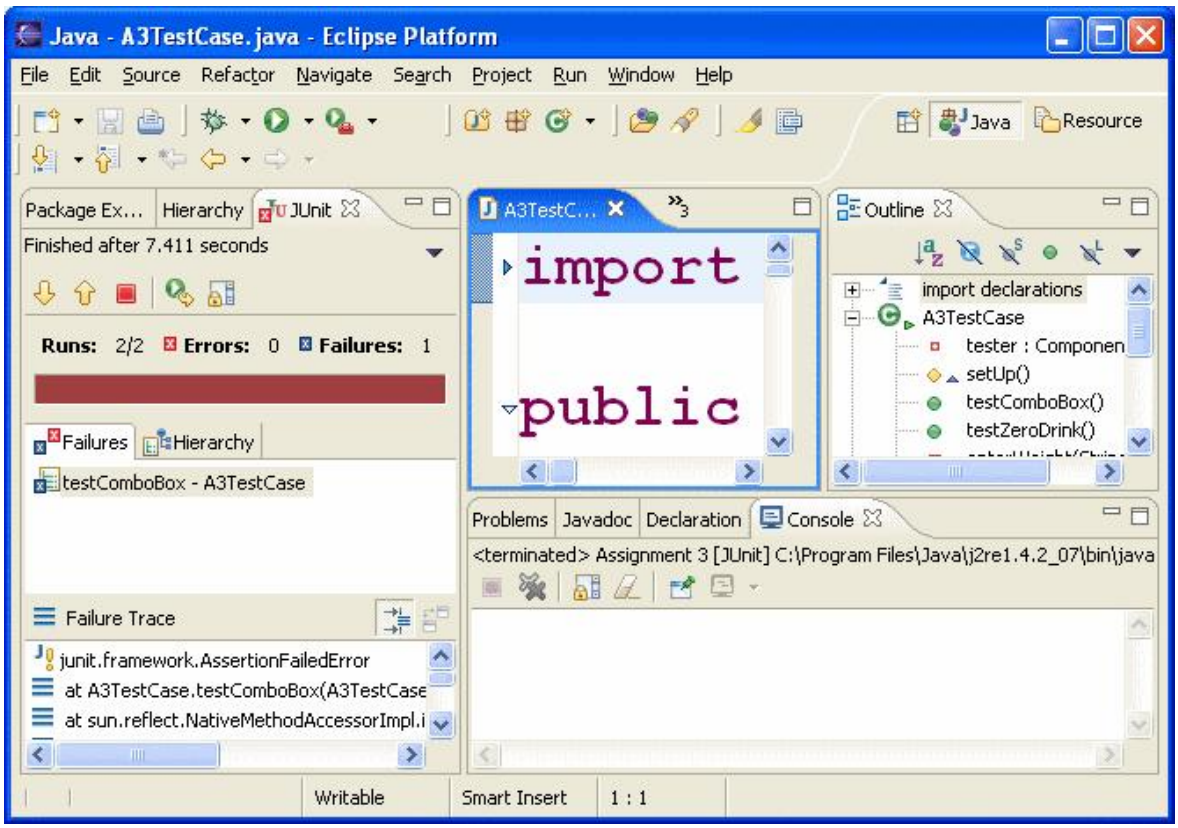
```
import junit.framework.TestCase;
public class RectangleTest
  extends TestCase {

  public void testQ2(){
    Rectangle r = newRect();
    r.setheight(2);
    r.setwidth(3);
    assertEquals(6, r.area());
  }

  protected Rectangle newRect(){
    return new Rectangle(1,1);
  }
}
```

When the test cases are executed, testQ2 in RectangleTest passes, but testQ2 in SquareTest fails. Explain what the problem is, and how you would fix it.

ANSWER:

[Answer space for question 2]

3. **[8 marks]** Imagine you are hired by Eclipse's testing team. Describe 2 GUI test cases, and 2 functionality test cases for Eclipse. As a reminder, following is part of a screendump of the GUI of Eclipse. You do not have to derive your test cases from the screendump.



ANSWER:

4. **[18 marks]** Consider a logic function Z of four boolean variables A, B, C, D. The formula for Z is

$$Z = AB + BC + \sim A \sim B \sim CD$$

Describe the process of applying the Binary Decision Diagram Determinants strategy and the Variable Negation strategy in this case. Produce a minimum size test suite for the Variable Negation strategy. Identify clearly the two test suites derived. Briefly discuss the differences between the two test suites.

ANSWER:

[Answer space for question 4]

5. **[12 marks]** Apply the Category-Partition test pattern to the following method:

   I, J, and K are integers.
   The method calculates K = IntegerPartOf (SquareRoot (I * J)).

   You need only consider cases in which you enter integer values into I and J.

   ANSWER:

6. **[22 marks]** Consider the following Java piece of code.

```
for (int i=1;i<5;i++) {
  if (rand(1,5) == i) break;
}
for (int j=1;j<5;j++) {
  System.out.println(j);
}
int x = rand(1,4);
switch (x) {
  case 1: System.out.println(x); break;
  case 2: System.out.println(x-23); break;
  case 3: System.out.println(x+42); break;
  case 4: System.out.println(x*3); break;
}
if (x>2) {
  for (int k=1;k<x;k++) {
    if (k == rand(1,4)) break;
  }
}
```

Method `rand(int,int)` returns a random integer number from the interval specified by its arguments, e.g. `rand(1,4)` may return either 1, 2, 3, or 4.

Draw the Control Flow Graph for this piece of code.

How many paths are there through this piece of code? Please show and/or explain your calculations.

ANSWER:

[Answer space for question 6]

7. **[22 marks]** Following is the statechart for the Traffic Light Controller system we discussed in class (it can also be found on page 196 of the textbook). Show all steps of applying the N+ testing strategy to it. Clearly indicate the test suite derived.

   Ignore the PowerOn event on the transition from the default state to the Off state. Combinations of states and events not indicated in the statechart are meant to be ignored by this system.

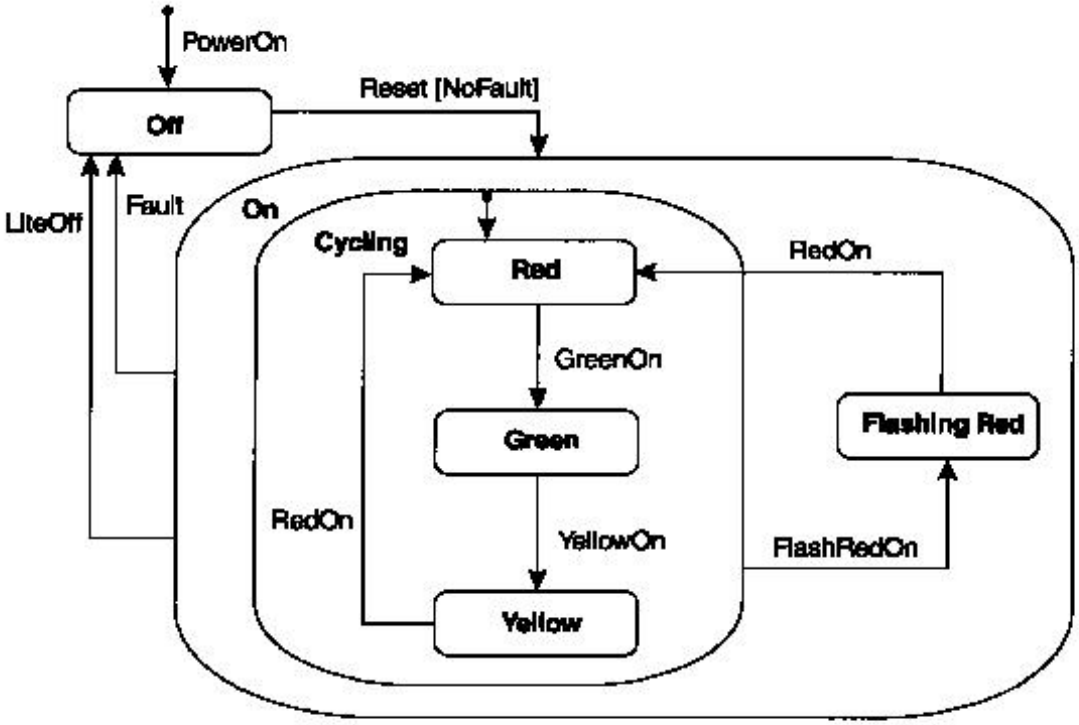


FIGURE 7.12 Statechart for traffic light.

ANSWER:

[Answer space for question 7]