

## 1. Part 1

The screenshot displays an IDE interface with a project named 'lab6'. The left sidebar shows the project structure, including a 'test' directory with a 'lab6' subdirectory containing 'BearingTest.java' and 'RectangleTest.java'. The main editor shows the code for 'BearingTest.java', which includes imports for JUnit and Hamcrest, and three test methods: 'answersValidBearing()', 'answersAngleBetweenItAndAnotherBearing()', and 'angleBetweenIsNegativeWhenThisBearingSmaller()'. The 'Run' button is highlighted next to the third method. The bottom status bar shows the execution results: 'Tests passed: 1 of 1 test - 7 ms'. The console output shows the command used to run the tests and the message 'Process finished with exit code 0'.

```
1 package lab6;
2 import org.junit.*;
3 import static org.hamcrest.CoreMatchers.*;
4 import static org.junit.Assert.*;
5
6 public class BearingTest {
7
8     @Test
9     public void answersValidBearing() throws Exception {
10         assertThat(new Bearing(Bearing.MAX).value(), equalTo(Bearing.MAX));
11     }
12
13     @Test
14     public void answersAngleBetweenItAndAnotherBearing() throws Exception {
15         assertThat(new Bearing(value: 15).angleBetween(new Bearing(value: 12)), equalTo(
16     }
17
18     @Test
19     public void angleBetweenIsNegativeWhenThisBearingSmaller() throws Exception {
20         assertThat(new Bearing(value: 12).angleBetween(new Bearing(value: 15)), equalTo(
21     }
22
23 }
```

Run: BearingTest.answersValidBearing x

Tests passed: 1 of 1 test - 7 ms

BearingTest (lab 7 ms) /usr/lib/jvm/java-11-openjdk/bin/java -ea -Didea.test.cyclic.buffer.size=1048576 -javaagent:/usr/

answersValid 7 ms

Process finished with exit code 0

a.



The screenshot shows an IDE with a project named 'lab6'. The left sidebar contains a file explorer with the following items: 'lab6', 'Bearing', 'BearingOutOfRangeException', 'BearingTest', 'ConstrainsSidesTo', 'Rectangle', 'RectangleTest', 'test', 'lab6', 'BearingTest.java', 'RectangleTest.java', 'classpath', 'project', 'lock.Lab 6 Document.docx#', 'hamcrest-core-1.3.jar', 'unit-4.13.2.jar', 'lab6.iml', 'lab6\_Book\_pages.pdf', 'lab 6 Document.docx', 'ernal Libraries', and 'atches and Consoles'. The main editor displays the code for 'BearingTest.java' with line numbers 37 to 77. The code includes several test methods: 'angleBetweenIsNegativeWhenThisBearingSmaller2()', 'angleBetweenIsNegativeWhenThisBearingSmaller3()', 'angleBetweenIsNegativeWhenThisBearingSmaller4()', 'angleBetweenIsNegativeWhenThisBearingSmaller5()', and 'angleBetweenIsNegativeWhenThisBearingSmaller6()'. Each method uses 'assertThat' to verify the result of 'angleBetween' and catches exceptions. The bottom status bar shows 'Run: BearingTest.angleBetweenIsNegativeWhenThisBearingSmaller5()'. The bottom toolbar includes icons for running, debugging, and other IDE functions. The bottom panel displays the execution results: 'Tests passed: 1 of 1 test - 6 ms', 'BearingTest (lab6) 6 ms', and 'angleBetweenIsNegativeWhenThisBearingSmaller5() 6 ms'. The process finished with exit code 0.

```
37 public void angleBetweenIsNegativeWhenThisBearingSmaller2() {
38     try {
39         assertThat(new Bearing((value: 5)).angleBetween(new Bearing((value: 15))), equalTo(operand: -10));
40     } catch (Exception e) {
41         e.printStackTrace();
42     }
43 }
44
45 @Test
46 public void angleBetweenIsNegativeWhenThisBearingSmaller3() {
47     try {
48         assertThat(new Bearing((value: 20)).angleBetween(new Bearing((value: 10))), equalTo(operand: 10));
49     } catch (Exception e) {
50         e.printStackTrace();
51     }
52 }
53 @Test
54 public void angleBetweenIsNegativeWhenThisBearingSmaller4() {
55     try {
56         assertThat(new Bearing((value: 25)).angleBetween(new Bearing((value: 37))), equalTo(operand: -12));
57     } catch (Exception e) {
58         e.printStackTrace();
59     }
60 }
61 @Test
62 public void angleBetweenIsNegativeWhenThisBearingSmaller5() {
63     try {
64         assertThat(new Bearing((value: 1)).angleBetween(new Bearing((value: 2))), equalTo(operand: -1));
65     } catch (Exception e) {
66         e.printStackTrace();
67     }
68 }
69
70 @Test
71 public void angleBetweenIsNegativeWhenThisBearingSmaller6() {
72     try {
73         assertThat(new Bearing((value: 7)).angleBetween(new Bearing((value: 29))), equalTo((-22)));
74     } catch (Exception e) {
75         e.printStackTrace();
76     }
77 }
```

Run: BearingTest.angleBetweenIsNegativeWhenThisBearingSmaller5() x

Tests passed: 1 of 1 test - 6 ms

BearingTest (lab6) 6 ms /usr/lib/jvm/java-11-openjdk/bin/java -ea -Didea.test.cyclic.buffer.size=1048576 -javaagent:...

angleBetweenIsNegativeWhenThisBearingSmaller5() 6 ms Process finished with exit code 0

2.

The screenshot shows an IDE with a project explorer on the left, a code editor in the center, and a test runner at the bottom.

**Project Explorer (Left):**

- lab6
  - Bearing
  - BearingOutOfRangeException
  - BearingTest
  - ConstrainsSidesTo
  - Rectangle
  - RectangleTest

**Code Editor (Center):**

```
71 public void angleBetweenIsNegativeWhenThisBearingSmaller6() {  
72     try {  
73         assertThat(new Bearing(value: 7).angleBetween(new Bearing(value: 29)), equalTo((-22));  
74     } catch (Exception e) {  
75         e.printStackTrace();  
76     }  
77 }  
78  
79 @Test  
80 public void angleBetweenIsNegativeWhenThisBearingSmaller7() {  
81     try {  
82         assertThat(new Bearing(value: 20).angleBetween(new Bearing(value: 15)), equalTo(operand: 5));  
83     } catch (Exception e) {  
84         e.printStackTrace();  
85     }  
86 }  
87  
88 @Test  
89 public void angleBetweenIsNegativeWhenThisBearingSmaller8() {  
90     try {  
91         assertThat(new Bearing(value: 100).angleBetween(new Bearing(value: 2)), equalTo(operand: 94));  
92     } catch (Exception e) {  
93         e.printStackTrace();  
94     }  
95 }  
96  
97  
98  
99  
100 }  
101
```

**Test Runner (Bottom):**

Run: BearingTest.angleBetweenIsNegativeWhenThisBearingSmaller8() X

Tests passed: 1 of 1 test - 5 ms

✓ BearingTest (lab6) 5 ms /usr/lib/jvm/java-11-openjdk/bin/java -ea -Didea.test.cyclic.buffer.size=1048576 -javaagent

✓ angleBetweenIsNegativeWhenThisBearingSmaller8() 5 ms Process finished with exit code 0

3. Part 3
  - a. Tests

The screenshot shows an IDE with a project structure on the left and a code editor on the right. The project structure includes a `src` directory with a `lab6` subdirectory containing `Bearing`, `BearingOutOfRange`, `BearingTest`, `ConstrainsSidesTo`, `Rectangle`, and `RectangleTest`. The `test` directory contains a `lab6` subdirectory with `BearingTest.java` and `RectangleTest.java`. The code editor shows the `RectangleTest` class with the following code:

```
import ...  
public class RectangleTest {  
    private Rectangle rectangle;  
    @After  
    public void ensureInvariant() { assertThat(rectangle, constrainsSidesTo(10));  
    @Test  
    public void answersArea() {  
        18 rectangle = new Rectangle(new Point(x: 5, y: 5), new Point (x: 19, y: 19));  
        19 assertThat(rectangle.area(), equalTo(operand: 50));  
        20 }  
        21  
        22 @Test  
        23 public void allowsDynamicallyChangingSize() {  
        24     rectangle = new Rectangle(new Point(x: 5, y: 5));  
        25     rectangle.setOppositeCorner(new Point(x: 130, y: 130));  
        26     assertThat(rectangle.area(), equalTo(operand: 15625));  
        27 }  
        28 }
```

The Run window shows the execution of `RectangleTest.answersArea`. The test failed with the following error:

```
Tests failed: 1 of 1 test - 31 ms  
RectangleTest ( 31 ms)  
answersAre: 31 ms  
java.lang.AssertionError:  
Expected: <50>  
but: was <70>  
Expected :<50>  
Actual   :<70>  
<Click to see difference>  
at org.hamcrest.MatcherAssert.assertThat(MatcherAssert.java:20) <2 internal lines>  
at lab6.RectangleTest.answersArea(RectangleTest.java:19) <28 internal lines>  
Process finished with exit code 255
```

i.

The screenshot shows an IDE with a project structure on the left and a code editor on the right. The project structure includes a 'src' directory with a 'lab6' package containing 'RectangleTest.java'. The code editor shows the following Java code:

```
10 import java.awt.*;
11 public class RectangleTest {
12     private Rectangle rectangle;
13     @After
14     public void ensureInvariant() {
15         assertEquals("rectangle area is 50", rectangle.area(), 50);
16     }
17     @Test
18     public void answersArea() {
19         rectangle = new Rectangle(new Point(5, 5), new Point(19, 19));
20         assertEquals("rectangle area is 50", rectangle.area(), 50);
21     }
22     @Test
23     public void allowsDynamicallyChangingSize() {
24         rectangle = new Rectangle(new Point(5, 5), new Point(130, 130));
25         rectangle.setOppositeCorner(new Point(130, 130));
26         assertEquals("rectangle area is 15625", rectangle.area(), 15625);
27     }
28 }
```

Below the code editor, the 'Run' tab shows the test results for 'RectangleTest.allowsDynamicallyChangingSize'. The test failed with the following error:

```
Tests failed: 1 of 1 test - 88 ms
RectangleTest ( 88 ms)
  allowsDynamicallyChangingSize ( 88 ms)
    java.lang.AssertionError:
      Expected: both sides must be <= 100
      but: was <Rectangle(origin java.awt.Point[x=5,y=5] opposite java.awt.Point[x=130,y=130])>
      Expected: both sides must be <= 100
      Actual   :<Rectangle(origin java.awt.Point[x=5,y=5] opposite java.awt.Point[x=130,y=130])>
      <Click to see difference>
      at org.hamcrest.MatcherAssert.assertThat(MatcherAssert.java:20) <2 internal lines>
      at lab6.RectangleTest.ensureInvariant(RectangleTest.java:14) <28 internal lines>
    Process finished with exit code 255
```

ii.

#### 4. Fix Tests

The screenshot shows an IDE with a project structure on the left. The 'src' folder contains a 'lab6' subfolder with several test classes, including 'RectangleTest'. The 'test' folder also contains a 'lab6' subfolder with 'BearingTest.java' and 'RectangleTest.java'. The main editor displays the code for 'RectangleTest.java'. The code includes a private 'Rectangle' field, an '@After' method 'ensureInvariant()' that asserts the rectangle's sides are within a length of 100, and two '@Test' methods. The 'answersArea()' method creates a 'Rectangle' with opposite corners at (5, 5) and (19, 10), and asserts its area is 70. The 'allowsDynamicallyChangingSize()' method creates a 'Rectangle' with opposite corners at (5, 5) and (99, 99), and asserts its area is 8836. The bottom status bar indicates that the test 'RectangleTest.answersArea' passed in 16 ms.

```

10 public class RectangleTest {
11     private Rectangle rectangle;
12     @After
13     public void ensureInvariant() { assertThat(rectangle, constrainsSidesTo(length: 100)); }
14
15     @Test
16     public void answersArea() {
17         rectangle = new Rectangle(new Point(x: 5, y: 5), new Point(x: 19, y: 10));
18         assertThat(rectangle.area(), equalTo(operand: 70));
19     }
20
21     @Test
22     public void allowsDynamicallyChangingSize() {
23         rectangle = new Rectangle(new Point(x: 5, y: 5));
24         rectangle.setOppositeCorner(new Point(x: 99, y: 99));
25         assertThat(rectangle.area(), equalTo(operand: 8836));
26     }
27 }
28

```

Run: RectangleTest.answersArea X  
 Tests passed: 1 of 1 test - 16 ms  
 RectangleTest (16 ms)  
 answersArea: 16 ms  
 Process finished with exit code 0

a.

This screenshot is similar to the previous one, showing the same IDE and code. However, the bottom status bar now indicates that the test 'RectangleTest.allowsDynamicallyChangingSize' passed in 17 ms.

```

12     @After
13     public void ensureInvariant() { assertThat(rectangle, constrainsSidesTo(length: 100)); }
14
15     @Test
16     public void answersArea() {
17         rectangle = new Rectangle(new Point(x: 5, y: 5), new Point(x: 19, y: 10));
18         assertThat(rectangle.area(), equalTo(operand: 70));
19     }
20
21     @Test
22     public void allowsDynamicallyChangingSize() {
23         rectangle = new Rectangle(new Point(x: 5, y: 5));
24         rectangle.setOppositeCorner(new Point(x: 99, y: 99));
25         assertThat(rectangle.area(), equalTo(operand: 8836));
26     }
27 }
28

```

Run: RectangleTest.allowsDynamicallyChangingSize X  
 Tests passed: 1 of 1 test - 17 ms  
 RectangleTest (17 ms)  
 allowsDynar: 17 ms  
 Process finished with exit code 0

b.

5.

- a. A throw exception throws a user-defined exception and allows a “fallback” when the code fails and proceeds to be an expected error when that exception occurs. This is why it fixes the code, because it allows the code to run even though there might be a potential problem, and allows it to be caught without causing any other problem related to the exception.

- b. A try-catch method is code that also allows “fallback” but runs the code that is in the try method and if any exception is caught then it runs the code within the catch method. It fixes the code, by allowing you to run a separate part of the code when an expected error occurs. This allows the code to pass.
- c. The try-catch will execute code that can throw exceptions, and use that exception to run additional code. Meanwhile, throwing an exception will allow you to throw an exception of a specific type but won't allow you to run additional code when it's thrown.