



Lab Testing 2 – CSE332

Software Testing and Validation

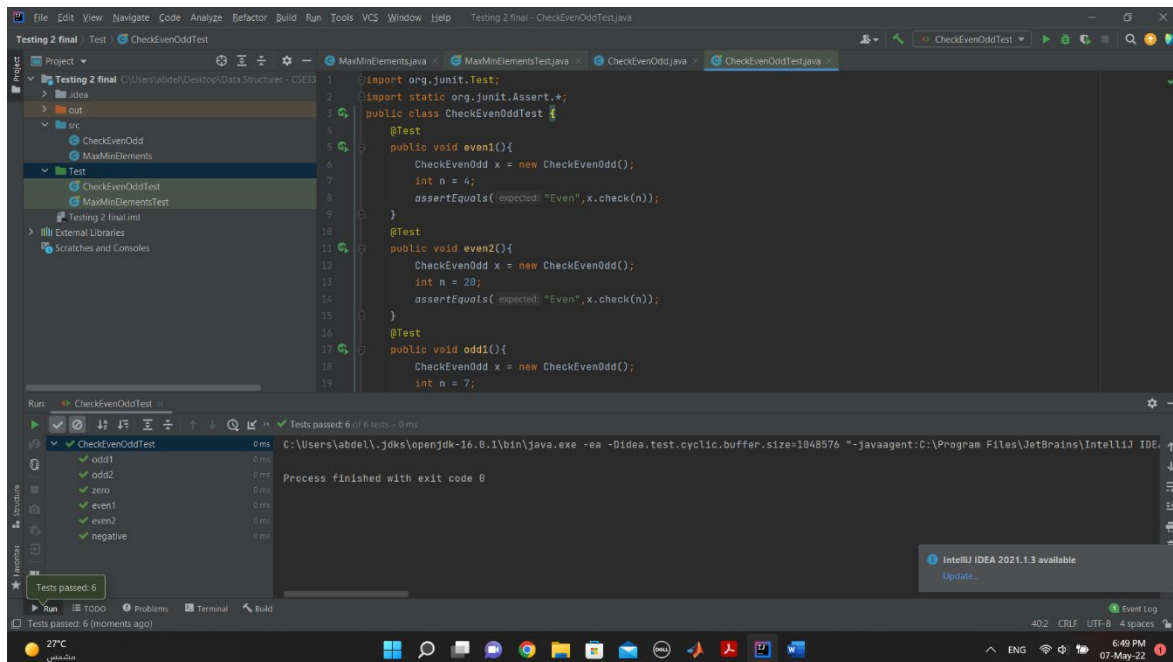
Abdelrahman Ibrahim Mohamed Ismail –
19P3049

Question 1.A Code:

```
public class CheckEvenOdd {
    public String check(int n) {
        if (n < 0)
            throw new IllegalArgumentException("Weight must be greater than 0!");
        if (n % 2 == 0)
            return "Even";
        else
            return "Odd";
    }
}
```

```
import org.junit.Test;
import static org.junit.Assert.*;
public class CheckEvenOddTest {
    @Test
    public void even1() {
        CheckEvenOdd x = new CheckEvenOdd();
        int n = 4;
        assertEquals("Even", x.check(n));
    }
    @Test
    public void even2() {
        CheckEvenOdd x = new CheckEvenOdd();
        int n = 20;
        assertEquals("Even", x.check(n));
    }
    @Test
    public void odd1() {
        CheckEvenOdd x = new CheckEvenOdd();
        int n = 7;
        assertEquals("Odd", x.check(n));
    }
    @Test
    public void odd2() {
        CheckEvenOdd x = new CheckEvenOdd();
        int n = 11;
        assertEquals("Odd", x.check(n));
    }
    @Test
    public void zero() {
        CheckEvenOdd x = new CheckEvenOdd();
        int n = 0;
        assertEquals("Even", x.check(n));
    }
    @Test(expected = IllegalArgumentException.class)
    public void negative() {
        CheckEvenOdd x = new CheckEvenOdd();
        int n = -4;
        x.check(n);
    }
}
```

Output:



Question 1.B code:

```
public class MaxMinElements { public int Max(int[] x){
    int maxElement=x[0];
    for(int i=0; i<x.length; i++){
        if(x[i]>maxElement){
            maxElement=x[i];
        }
    }
    return maxElement;
}

    public int Min(int[] x){
        int minElement=x[0];
        for(int i=0; i<x.length; i++){
            if(x[i]<minElement){
                minElement=x[i];
            }
        }
        return minElement;
    }
}

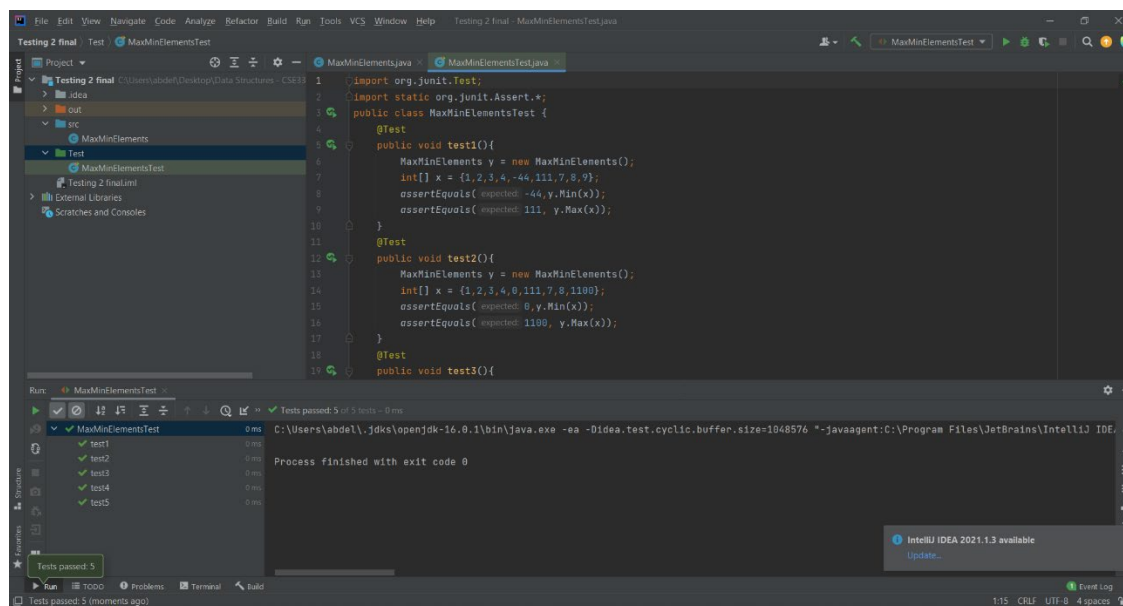
import org.junit.Test;
import static org.junit.Assert.*;
public class MaxMinElementsTest {
    @Test
    public void test1(){
        MaxMinElements y = new MaxMinElements();
        int[] x = {1,2,3,4,-44,111,7,8,9};
        assertEquals(-44,y.Min(x));
    }
}
```

```

        assertEquals(111, y.Max(x));
    }
    @Test
    public void test2(){
        MaxMinElements y = new MaxMinElements();
        int[] x = {1,2,3,4,0,111,7,8,1100};
        assertEquals(0,y.Min(x));
        assertEquals(1100, y.Max(x));
    }
    @Test
    public void test3(){
        MaxMinElements y = new MaxMinElements();
        int[] x = {8,9};
        assertEquals(8,y.Min(x));
        assertEquals(9, y.Max(x));
    }
    @Test
    public void test4(){
        MaxMinElements y = new MaxMinElements();
        int[] x = {1};
        assertEquals(1,y.Min(x));
        assertEquals(1, y.Max(x));
    }
    @Test
    public void test5(){
        MaxMinElements y = new MaxMinElements();
        int[] x = {65, 87, 1, 66, 88, 112, 210, -1};
        assertEquals(-1,y.Min(x));
        assertEquals(210, y.Max(x));
    }
}

```

Output:



Question 2

```
import org.junit.jupiter.api.BeforeEach; import org.junit.jupiter.api.Nested;
import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.*;

class StateChartTest {

    @Nested
    class EdgeCoverage{

        StateChart mainObj;

        @BeforeEach
        public void setup(){
            mainObj = new StateChart();
        }

        @Test
        public void ECtest1(){
            String[] res = mainObj.input('a'); assertEquals("NORMAL",
res[0]); assertEquals("DATE", res[1]); assertEquals("2000-1-1", res[2]);
            assertEquals("0:0", res[3]);
        }

        @Test
        public void ECtest2(){ mainObj.input('a');
            String[] res = mainObj.input('a'); assertEquals("NORMAL",
res[0]); assertEquals("TIME", res[1]); assertEquals("2000-1-1", res[2]);
            assertEquals("0:0", res[3]);
        }

        @Test
        public void ECtest3(){ mainObj.input('a');
            mainObj.input('a');
            String[] res = mainObj.input('c'); assertEquals("UPDATE",
res[0]); assertEquals("min", res[1]); assertEquals("2000-1-1", res[2]);
            assertEquals("0:0", res[3]);
        }

        @Test
        public void ECtest4(){ mainObj.input('a');
            mainObj.input('a');
            mainObj.input('c');

            String[] res = mainObj.input('b'); assertEquals("UPDATE",
res[0]); assertEquals("min", res[1]); assertEquals("2000-1-1", res[2]);
            assertEquals("0:1", res[3]);
        }

        @Test
        public void ECtest5(){ mainObj.input('a');
            mainObj.input('a');
            mainObj.input('c');
            mainObj.input('b');
```

```

        String[] res = mainObj.input('a'); assertEquals("UPDATE",
res[0]); assertEquals("hour", res[1]); assertEquals("2000-1-1", res[2]);
        assertEquals("0:1", res[3]);
    }

    @Test
    public void ECtest6(){ mainObj.input('a');
        mainObj.input('a');
        mainObj.input('c');
        mainObj.input('b');
        mainObj.input('a');
        String[] res = mainObj.input('b'); assertEquals("UPDATE",
res[0]); assertEquals("hour", res[1]); assertEquals("2000-1-1", res[2]);
        assertEquals("1:1", res[3]);
    }

    @Test
    public void ECtest7(){ mainObj.input('a');
        mainObj.input('a');
        mainObj.input('c');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        String[] res = mainObj.input('a'); assertEquals("UPDATE",
res[0]); assertEquals("day", res[1]); assertEquals("2000-1-1", res[2]);
        assertEquals("1:1", res[3]);
    }

    @Test
    public void ECtest8(){ mainObj.input('a');
        mainObj.input('a');
        mainObj.input('c');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');

        mainObj.input('a');
        String[] res = mainObj.input('b'); assertEquals("UPDATE",
res[0]); assertEquals("day", res[1]); assertEquals("2000-1-2", res[2]);
        assertEquals("1:1", res[3]);
    }

    @Test
    public void ECtest9(){ mainObj.input('a');
        mainObj.input('a');
        mainObj.input('c');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        String[] res = mainObj.input('a'); assertEquals("UPDATE",
res[0]); assertEquals("month", res[1]); assertEquals("2000-1-2", res[2]);
        assertEquals("1:1", res[3]);
    }

    @Test

```

```

    public void ECtest10() { mainObj.input('a');
        mainObj.input('a');
        mainObj.input('c');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        String[] res = mainObj.input('b'); assertEquals("UPDATE",
res[0]); assertEquals("month", res[1]); assertEquals("2000-2-2", res[2]);
        assertEquals("1:1", res[3]);
    }

    @Test
    public void ECtest11() { mainObj.input('a');
        mainObj.input('a');
        mainObj.input('c');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        String[] res = mainObj.input('a');
        assertEquals("UPDATE", res[0]);

        assertEquals("year", res[1]);
        assertEquals("2000-2-2", res[2]);
        assertEquals("1:1", res[3]);
    }

    @Test
    public void ECtest12() { mainObj.input('a');
        mainObj.input('a');
        mainObj.input('c');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        String[] res = mainObj.input('b'); assertEquals("UPDATE",
res[0]); assertEquals("year", res[1]); assertEquals("2001-2-2", res[2]);
        assertEquals("1:1", res[3]);
    }

    @Test
    public void ECtest13() { mainObj.input('a');
        mainObj.input('a');
        mainObj.input('c');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');

```

```

        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        String[] res = mainObj.input('a'); assertEquals("NORMAL",
res[0]); assertEquals("TIME", res[1]); assertEquals("2001-2-2", res[2]);
        assertEquals("1:1", res[3]);
    }

    @Test
    public void ECtest14(){ mainObj.input('a');
        mainObj.input('a');
        mainObj.input('c');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');

        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        String[] res = mainObj.input('c'); assertEquals("UPDATE",
res[0]); assertEquals("min", res[1]); assertEquals("2001-2-2", res[2]);
        assertEquals("1:1", res[3]);
    }

    @Test
    public void ECtest15(){ mainObj.input('a');
        mainObj.input('a');
        mainObj.input('c');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('c');
        String[] res = mainObj.input('d'); assertEquals("NORMAL",
res[0]); assertEquals("TIME", res[1]); assertEquals("2001-2-2", res[2]);
        assertEquals("1:1", res[3]);
    }

    @Test
    public void ECtest16(){ mainObj.input('a');
        mainObj.input('a');
        mainObj.input('c');
        mainObj.input('b');
        mainObj.input('a');

```



```

        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('c');
        mainObj.input('d');
        String[] res = mainObj.input('b'); assertEquals("ALARM", res[0]);
assertEquals("Alarm", res[1]); assertEquals("2001-2-2", res[2]);
        assertEquals("1:1", res[3]);
    }

```

```

@Test
public void ECtest17(){ mainObj.input('a');
    mainObj.input('a');
    mainObj.input('c');
    mainObj.input('b');
    mainObj.input('a');
    mainObj.input('b');
    mainObj.input('a');
    mainObj.input('b');
    mainObj.input('a');
    mainObj.input('b');
    mainObj.input('a');
    mainObj.input('b');
    mainObj.input('a');
    mainObj.input('c');
    mainObj.input('d');
    mainObj.input('b');
    String[] res = mainObj.input('a'); assertEquals("ALARM", res[0]);
assertEquals("Chime", res[1]); assertEquals("2001-2-2", res[2]);
    assertEquals("1:1", res[3]);
}

```

```

@Test
public void ECtest18(){ mainObj.input('a');
    mainObj.input('a');
    mainObj.input('c');
    mainObj.input('b');
    mainObj.input('a');
    mainObj.input('b');
    mainObj.input('a');
    mainObj.input('b');
    mainObj.input('a');
    mainObj.input('b');
    mainObj.input('a');
    mainObj.input('b');
    mainObj.input('a');
    mainObj.input('c');
    mainObj.input('d');
    mainObj.input('b');
    mainObj.input('a');
    String[] res = mainObj.input('d'); assertEquals("NORMAL",

```

```

res[0]); assertEquals("TIME", res[1]); assertEquals("2001-2-2", res[2]);
    assertEquals("1:1", res[3]);
}

}

@Nested class ADUP{

    StateChart mainObj;

    @BeforeEach
    public void setup(){
        mainObj = new StateChart();
    }

    @Test
    public void ADUPTest1(){
        String[] res = mainObj.input('c'); assertEquals("UPDATE",
res[0]); assertEquals("min", res[1]); assertEquals("2000-1-1", res[2]);
        assertEquals("0:0", res[3]);
    }

    @Test
    public void ADUPTest2(){ mainObj.input('c');
        String[] res = mainObj.input('b'); assertEquals("UPDATE",
res[0]); assertEquals("min", res[1]); assertEquals("2000-1-1", res[2]);
        assertEquals("0:1", res[3]);
    }

    @Test
    public void ADUPTest3(){ mainObj.input('c');
        mainObj.input('b');
        String[] res = mainObj.input('b'); assertEquals("UPDATE",
res[0]); assertEquals("min", res[1]); assertEquals("2000-1-1", res[2]);
        assertEquals("0:2", res[3]);
    }

    @Test
    public void ADUPTest4(){ mainObj.input('c');
        mainObj.input('b');
        mainObj.input('b');
        String[] res = mainObj.input('a'); assertEquals("UPDATE",
res[0]); assertEquals("hour", res[1]); assertEquals("2000-1-1", res[2]);
        assertEquals("0:2", res[3]);
    }

    @Test
    public void ADUPTest5(){ mainObj.input('c');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        String[] res = mainObj.input('b');

        assertEquals("UPDATE", res[0]); assertEquals("hour", res[1]);
assertEquals("2000-1-1", res[2]);

```

```

        assertEquals("1:2", res[3]);
    }

    @Test
    public void ADUPTest6(){ mainObj.input('c');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        String[] res = mainObj.input('b'); assertEquals("UPDATE",
res[0]); assertEquals("hour", res[1]); assertEquals("2000-1-1", res[2]);
        assertEquals("2:2", res[3]);
    }

    @Test
    public void ADUPTest7(){ mainObj.input('c');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        String[] res = mainObj.input('a'); assertEquals("UPDATE",
res[0]); assertEquals("day", res[1]); assertEquals("2000-1-1", res[2]);
        assertEquals("2:2", res[3]);
    }

    @Test
    public void ADUPTest8(){ mainObj.input('c');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        String[] res = mainObj.input('b'); assertEquals("UPDATE",
res[0]); assertEquals("day", res[1]); assertEquals("2000-1-2", res[2]);
        assertEquals("2:2", res[3]);
    }

    @Test
    public void ADUPTest9(){ mainObj.input('c');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');

        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        String[] res = mainObj.input('b'); assertEquals("UPDATE",
res[0]); assertEquals("day", res[1]); assertEquals("2000-1-3", res[2]);
        assertEquals("2:2", res[3]);
    }

    @Test
    public void ADUPTest10(){ mainObj.input('c');
        mainObj.input('b');

```

```

        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        String[] res = mainObj.input('a'); assertEquals("UPDATE",
res[0]); assertEquals("month", res[1]); assertEquals("2000-1-3", res[2]);
        assertEquals("2:2", res[3]);
    }

    @Test
    public void ADUPTest11(){ mainObj.input('c');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        String[] res = mainObj.input('b'); assertEquals("UPDATE",
res[0]); assertEquals("month", res[1]); assertEquals("2000-2-3", res[2]);
        assertEquals("2:2", res[3]);
    }

    @Test
    public void ADUPTest12(){ mainObj.input('c');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');

        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        String[] res = mainObj.input('b'); assertEquals("UPDATE",
res[0]); assertEquals("month", res[1]); assertEquals("2000-3-3", res[2]);
        assertEquals("2:2", res[3]);
    }

    @Test
    public void ADUPTest13(){ mainObj.input('c');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');

```

```

        mainObj.input('b');
        mainObj.input('b');
        String[] res = mainObj.input('a'); assertEquals("UPDATE",
res[0]); assertEquals("year", res[1]); assertEquals("2000-3-3", res[2]);
        assertEquals("2:2", res[3]);
    }

    @Test
    public void ADUPTest14(){ mainObj.input('c');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        String[] res = mainObj.input('b'); assertEquals("UPDATE",
res[0]); assertEquals("year", res[1]); assertEquals("2001-3-3", res[2]);
        assertEquals("2:2", res[3]);
    }

    @Test
    public void ADUPTest15(){ mainObj.input('c');

        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        mainObj.input('b');
        mainObj.input('a');
        mainObj.input('b');
        String[] res = mainObj.input('b');
        assertEquals("UPDATE", res[0]);
        assertEquals("year", res[1]);
        assertEquals("2002-3-3", res[2]);
        assertEquals("2:2", res[3]);
    }

}
}

```

Output:

