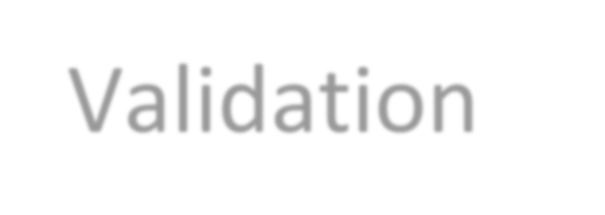
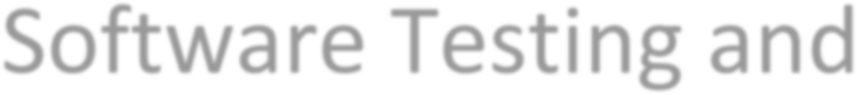


Lab Assignment 2 – CSE332



Software Testing and

Validation

Mohamed Hany El-Sayed Mohamed (19p6696)

Question 1A Even and Odd Code:

public class *CheckEvenOdd* {  
 public *String* check(int *no*){  
 if ( *no* < 0)  
 throw new IllegalArgumentException("Weight have to be higher than 0");  
 if(*no*%2==0)  
 return "weight is even";  
 else  
 return "weight is odd";  
 }  
}

## Test Cases:

import static *org.junit.Assert*.\*;  
import *org.junit.Test*;  
public class *CheckEvenOddTest* {  
 *@Test* public void even1(){  
 *CheckEvenOdd* y = new CheckEvenOdd(); int n = 16; *assertEquals*("Even",y.check(n));  
 }  
 *@Test* public void even2(){  
 *CheckEvenOdd* y = new CheckEvenOdd(); int n = 50; *assertEquals*("Even",y.check(n));  
 }  
 *@Test* public void odd1(){  
 *CheckEvenOdd* y = new CheckEvenOdd(); int n = 39; *assertEquals*("Odd",y.check(n));  
 }  
 *@Test* public void odd2(){  
 *CheckEvenOdd* y = new CheckEvenOdd(); int n = 51; *assertEquals*("Odd",y.check(n));  
 }  
 *@Test* public void zero(){  
 *CheckEvenOdd* y = new CheckEvenOdd();  
 int x = 0;  
 *assertEquals*("Even",y.check(x));  
 }  
 *@Test*(expected = *IllegalArgumentException*.class) public void negative(){  
 *CheckEvenOdd* n = new CheckEvenOdd(); int x = -8;  
 n.check(x);  
 }  
}

Output:

A screenshot of a computer

Description automatically generated with medium confidence

# Question 1B

Min and Max Elements

## Code:

public class *MinMaxElements*{ public int Max(int[] *m*){  
 int maxElement=*m*[0];  
 for(int i=0; i<*m*.length; i++)  
 { if(*m*[i]>maxElement){  
 maxElement=*m*[i];}  
 }  
 return maxElement;  
}  
 public int Min(int[]*n* )  
 { int minElement=*n*[0];  
 for(int i=0; i<*n*.length; i++)  
 { if(*n*[i]<minElement){  
 minElement=*n*[i];}  
 }  
 return minElement;  
 }  
}

Test Cases:

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();

import static *org.junit.Assert*.\*;  
import *org.junit.Test*;  
public class *MinMaxElementsTest* {  
 *@Test* public void testcase1(){  
 *MinMaxElements* y = new MinMaxElements(); int[] x ={423,43,5,6,764,7,865,-23,432,56,4,66};  
 *assertEquals*(-23,y.Min(x)); *assertEquals*(764, y.Max(x));  
 }  
 *@Test* public void testcase2(){  
 *MinMaxElements* y = new MinMaxElements(); int[] x = {434,543,5,54,64,76,86,-234,53,-34};  
 *assertEquals*(-234,y.Min(x)); *assertEquals*(543, y.Max(x));  
 }  
 *@Test* public void testcase3(){  
 *MinMaxElements* y = new MinMaxElements();  
 int[] x = {1,9}; *assertEquals*(1,y.Min(x)); *assertEquals*(9, y.Max(x));  
 }  
 *@Test* public void testcase4(){  
 *MinMaxElements* y = new MinMaxElements(); int[] x = {4}; *assertEquals*(4,y.Min(x)); *assertEquals*(4, y.Max(x));  
 }  
 *@Test* public void testcase5(){  
 *MinMaxElements* y = new MinMaxElements(); int[] x = {53, 345, 42, 532, 421, 23, 2022, -24};  
 *assertEquals*(-24,y.Min(x)); *assertEquals*(2022, y.Max(x));  
 }  
}

Output:

A screenshot of a computer

Description automatically generated with medium confidence

# Question 2

## Code:

public class StateChart {

public String state = "NORMAL"; public String state1 = "TIME";

public int m=0, h=0, D=1, M=1, Y=2000;

public String[] input(char c){

if(state.equals("NORMAL")) { if (c == 'c') {

state = "UPDATE"; state1 = "min";

}

if (c == 'b') { state = "ALARM";

state1 = "Alarm";

}

if (c == 'a') {

if (state1.equals("TIME")) { state1 = "DATE";

} else {

state1 = "TIME";

}

}

}

if(state.equals("UPDATE")) { if (c == 'd') {

state = "NORMAL"; state1 = "TIME";

}

if (c == 'a') {

if (state1.equals("year")) { state = "NORMAL";

state1 = "TIME";

}

if (state1.equals("month")) { state1 = "year";

}

if (state1.equals("day")) { state1 = "month";

}

if (state1.equals("hour")) { state1 = "day";

}

if (state1.equals("min") ) { state1 = "hour";

}

}

if (c == 'b') {

if (state1.equals("min")) { m++;

if (m >= 60) { m = 0;

}

}

if (state1.equals("hour")) { h++;

if (h >= 24) { h = 0;

}

}

if (state1.equals("day")) { D++;

if (D > 31) { D = 1;

}

}

if (state1.equals("month")) { M++;

if (M > 12) { M = 1;

}

}

if (state1.equals("year")) { Y++;

if (Y > 2100) { Y = 2000;

}

}

}

}

if(state.equals("ALARM")){ if (c == 'd') {

state = "NORMAL"; state1 = "TIME";

}

if (c == 'a') { state1 = "Chime";

}

}

return new String[]{state, state1, String.*valueOf*(Y) + "-" + String.*valueOf*(M) + "-" + String.*valueOf*(D), String.*valueOf*(h) + ":" + String.*valueOf*(m) };

}

}

## Tests:

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:

