



CS265FZ Software Testing
Lab 2 – Boundary Value Testing

There are THREE exercises to be completed.

Two pieces of work need to be submitted:

1. Fill in this lab sheet and submit it to Moodle. You don't need to attach your source code in this form. You need to upload your source code separately.
2. Submit all the required source code to Moodle. Make sure your source code is tested in Eclipse and is executable.
3. Make sure you provide detailed comments in the source code:
 - a. Identify the fault(s) in the source code
 - b. How did you fixed the fault(s)?

Problem 1

(Source Code: *Lab2_Program1.java*) A store in a city offers different discounts depending on the purchases made by the individual. To test the software that calculates the discounts, it is possible to identify the ranges of purchase values that earn the different discounts. For example, a purchase in the range of €1 to €50 inclusive has no discount, a purchase over €50 to €200 has a 5% discount, and purchases of over €201 inclusive to €500 have a 10% discounts, and purchases of €501 and above have a 15% discount. Invalid input will have a return value of 0 as discount.

Note: discounts are presented as follows: no discount is presented with 0, discount of 5% with 0.05, discount of 10% with 0.1, and 15% discount is presented as 0.15.

Task 1

Identify the input boundary value classes and output boundary classes.

Input Boundary Values		
Parameter	Test Class	Boundary Value
purchase	BV-1	Integer.MIN_VALUE
	BV-2	0
	BV-3	1
	BV-4	50
	BV-5	51
	BV-6	200
	BV-7	201
	BV-8	500
	BV-9	501
	BV-10	Integer.Max_value

Output Boundary Values		
Parameter	Test Case	Boundary Value
Return Value (discount)	BV-11	0
	BV-12	0.05
	BV-13	0.1
	BV-14	0.15

Task 2

Fill in the test data table (The following is one possible solution to the problem. Students may have equally valid solutions that are different from the example solution below.)

Test ID	Test Cases/Boundaries Covered	Input	Expected Outputs
		<i>purchase</i>	<i>Discount</i>
T1.1	BV-1, 11	Integer.MIN_VALUE	0
T1.2	BV-2, [11]	0	0
T1.3	BV-3, [11]	1	0
T1.4	BV-4, [11]	50	0
T1.5	BV-5, 12	51	0.05
T1.6	BV-6, [12]	200	0.05
T1.7	BV-7, 13	201	0.1
T1.8	BV-8, [13]	500	0.1
T1.9	BV-9, 14	501	0.15
T1.10	BV-10, [14]	Integer.MAX_VALUE	0.15

Task 3

Based on the specification given above, write your testing code in JUnit 5 to test the source code of the program provided on Moodle (“*Lab2_Program1.java*”). Make sure your test code is named as “*Lab1_Task1.java*”.

Task 4

Based on the test results, provide the correct version of the “*Lab2_Program1.java*”, and rename it to “*Lab2_Program1_Fix.java*”.

Problem 2

The program (Source Code: *Lab2_Program2.java*) Grade combines an exam and coursework mark into a single grade. The values for exam and coursework are integers. If the exam or coursework mark is less than 50% then the grade returned is a 'Fail'. To pass the course with a 'Pass, C', the student must score between 50% and 60% in the exam, and at least 50% in the coursework. They will pass the course with 'Pass, B', if they score at least 60% in the exam and 50% in the coursework. In addition to this, if the average of the exam and the coursework is at least 70%, then they are awarded a 'Pass, A'. Input values that are less than 0 or greater than 100 for either the exam or coursework are invalid, and the program will return a message to say 'Marks out of range'.

Task 1

Identify the input boundary value classes and output boundary classes.

Input Boundary Values		
Parameter	Test Class	Boundary Value
exam	BV-1	Integer.MIN_VALUE
	BV-2	-1
	BV-3	0
	BV-4	49
	BV-5	50
	BV-6	59
	BV-7	60
	BV-8	69
	BV-9	70
	BV-10	100
	BV-11	101
	BV-12	Integer.MAX_VALUE
course	BV-13	Integer.MIN_VALUE
	BV-14	-1
	BV-15	0
	BV-16	49
	BV-17	50
	BV-18	69
	BV-19	70
	BV-20	100
	BV-21	101
	BV-22	Integer.MAX_VALUE

Output Boundary Values		
Parameter	Test Case	Boundary Value
Return Value (grade)	BV-23	Fail
	BV-24	C
	BV-25	B
	BV-26	A
	BV-27	Marks out of range

Task 2

Fill in the test data table (The following is one possible solution to the problem. Students may have equally valid solutions that are different from the example solution below.)

Test ID	Test Cases/ Boundaries Covered	Inputs		Expected Outputs
		<i>exam</i>	<i>course</i>	<i>Result</i>
T2.1	BV-1,13,27	Integer.MIN_VALUE	Integer.MIN_VALUE	Marks out of range
T2.2	BV-2,14,[27]	-1	-1	Marks out of range
T2.3	BV-3,15,23	0	0	Fail
T2.4	BV-[3],16,[23]	0	49	Fail
T2.5	BV-4, [15,23]	49	0	Fail
T2.6	BV-5, [16,23]	50	49	Fail
T2.7	BV-[4],17,24	49	50	C
T2.8	BV-[5],17,[24]	50	50	C
T2.9	BV-6, [17,24]	59	50	C
T2.10	BV-7, [17],25	60	50	B
T2.11	BV-8, [17,25]	69	50	B
T2.12	BV-[8],18,[25]	69	69	B
T2.13	BV-9,19,26	70	70	A
T2.14	BV-10,20,[26]	100	100	A
T2.15	BV-11,21,[27]	101	101	Marks out of range
T2.16	BV-12,22,[27]	Integer.MAX_VALUE	Integer.MAX_VALUE	Marks out of range

Task 3

Based on the specification given above, write your testing code in JUnit 5 to test the source code of the program provided on Moodle (“*Lab2_Program2.java*”). Make sure your test code is named as “*Lab2_Task2.java*”.

Task 4

Based on the test results, provide the correct version of the “*Lab2_Program2.java*”, and rename it to “*Lab2_Program2_Fix.java*”.

Problem 3

A program (Source Code: *Lab2_Program3.java*) is used to assess the cost of a car insurance policy. It takes three inputs of age, gender and marital status. If the age entered is less than 16 or greater than 65, the program returns a premium of zero. The input for gender takes the form of 'M' for male and 'F' for female. If an incorrect value for the gender is entered the program returns a premium of zero. In general a premium is €500. However, if a person is male, under 25 and not married then an extra €1500 is added to the premium. If the person is female or married the premium falls by €200, and if the person is aged between 45 and 65 inclusive the premium falls by €100.

Task 1

Identify the input boundary value classes and output boundary classes.

Input Boundary Values		
Parameter	Test Class	Boundary Value
Age	BV-1	15
	BV-2	16
	BV-3	24
	BV-4	25
	BV-5	44
	BV-6	45
	BV-7	65
	BV-8	66
gender	BV-9	M
	BV-10	F
	BV-11	X
married	BV-12	true
	BV-13	false

Output Boundary Values		
Parameter	Test Case	Boundary Value
Return Value (premium)	BV-14	0
	BV-15	200
	BV-16	300
	BV-17	400
	BV-18	500
	BV-19	2000

Task 2

Fill in the test data table (The following is one possible solution to the problem. Students may have equally valid solutions that are different from the example solution below.)

Test ID	Test Cases Covered	Inputs			Expected Outputs
		<i>age</i>	<i>gender</i>	<i>married</i>	<i>premium</i>
T3.1	BV-1,10,13,14	15	F	false	0
T3.2	BV-2, [10, 13],16	16	F	false	300
T3.3	BV-8,10,12,[14]	66	F	true	0
T3.4	BV-6, [10, 13],15	45	F	false	200
T3.5	BV-7, [10, 13, 15]	65	F	false	200
T3.6	BV-5, [10, 13, 16]	44	F	false	300
T3.7	BV-5,[9, 12, 16]	44	M	true	300
T3.8	BV-6, [9, 13],17	45	M	false	400
T3.9	BV-[7, 9, 13],17]	65	M	false	400
T3.10	BV-[4, 9, 13,18]	25	M	false	500
T3.11	BV-[4, 9, 13], 18	25	M	false	2000
T3.12	BV-[3, 9,13], 19	24	X	false	0

Task 3

Based on the specification given above, write your testing code in JUnit 5 to test the source code of the program provided on Moodle (“***Lab2_Program3.java***”). Make sure your test code is named as “***Lab2_Task3.java***”.

Task 4

Based on the test results, provide the correct version of the “***Lab2_Program3.java***”, and rename it to “***Lab2_Program3_Fix.java***”.