CS608
Software Testing

Dr. Stephen Brown

Room Eolas 116

stephen.brown@mu.ie

CS608

Testing with Boundary Value Analysis

(Essentials of Software Testing, Chapter 3)

Note: We will go through Boundary Value Analysis quite quickly – we’ve just done all the hard work for EP

Testin alysis

(Ess 3)

Introduction

- Equivalence Partition testing uses representative values from each range of values for which the specification states equivalent processing
- Programmers often make mistakes at the boundary values of these ranges, which will not be caught by equivalence partition testing as has been demonstrated in Fault 2
- We now introduce the black-box testing technique of boundary value analysis (BVA), starting with a worked example

Testing with Boundary Value Analysis

- Boundary values are the minimum and maximum values for each Equivalence Partition
- Having identified the partitions, identifying the boundary values is straightforward
- The goal is to verify that the software works correctly at these boundaries

Definition:

a boundary value is the value at the boundary of an equivalence partition. Each equivalence partition has exactly two boundary values.

Example

- Continue testing OnlineSales.giveDiscount()
- Summary – the method returns:

FULLPRICE if bonusPoints≤120 and not a goldCustomer

FULLPRICE if bonusPoints≤80 and a goldCustomer

DISCOUNT if bonusPoints>120

DISCOUNT if bonusPoints>80 and a goldCustomer

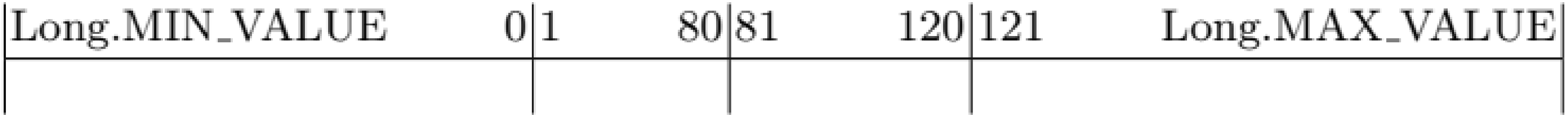
ERROR if any inputs are invalid (bonusPoints<1)

Step 1. Analysis

- Analyse the specification to identify the boundary values
- We will use the equivalence partitions previously identified for giveDiscount()
- If you are developing BVA from scratch, then you need to develop the value lines and partitions as shown for EP
- But if you have done EP first (which is usual) then you can reuse the value lines and partitions

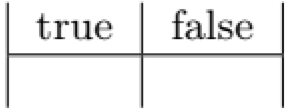
Specification-Based Ranges

- bonusPoints



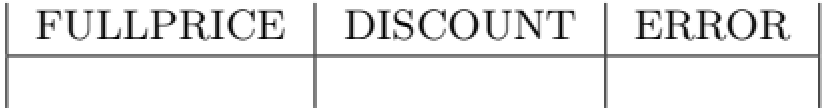
Long.MIN_VALUE 0	1 80	81 120	121 Long.MAX_VALUE

- goldCustomer



true	false

- Return Value



FULLPRICE	DISCOUNT	ERROR

Specification-Based Ranges

- bonusPoints

Long.MIN_VALUE 0	1 80	81 120	121 Long.MAX_VALUE

- goldCustomer

true	false

Single-Valued Partitions

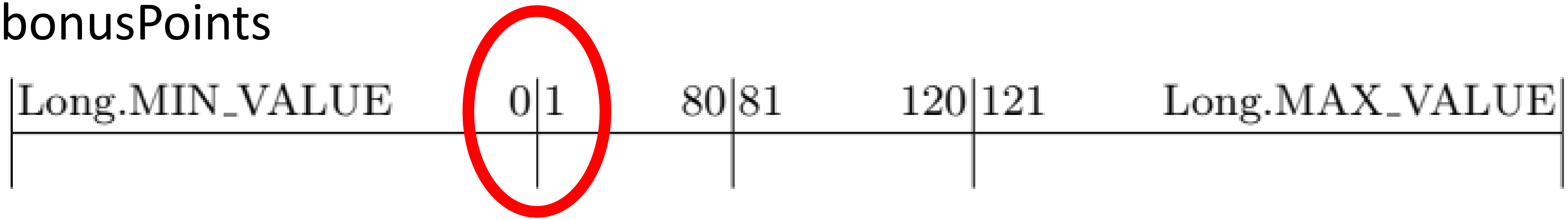
- Return Value

FULLPRICE	DISCOUNT	ERROR

Single-Valued Partitions

Specification-Based Ranges

- bonusPoints



LIKELY FAULT LOCATION

- goldCustomer

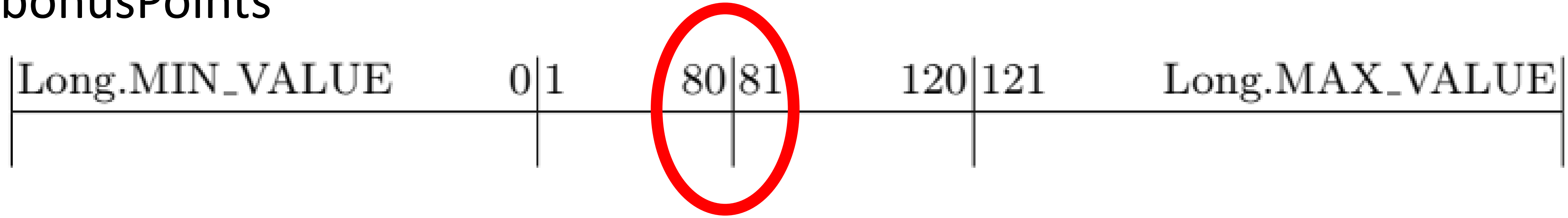
true	false

- Return Value

FULLPRICE	DISCOUNT	ERROR

Specification-Based Ranges

- bonusPoints



LIKELY FAULT LOCATION

- goldCustomer

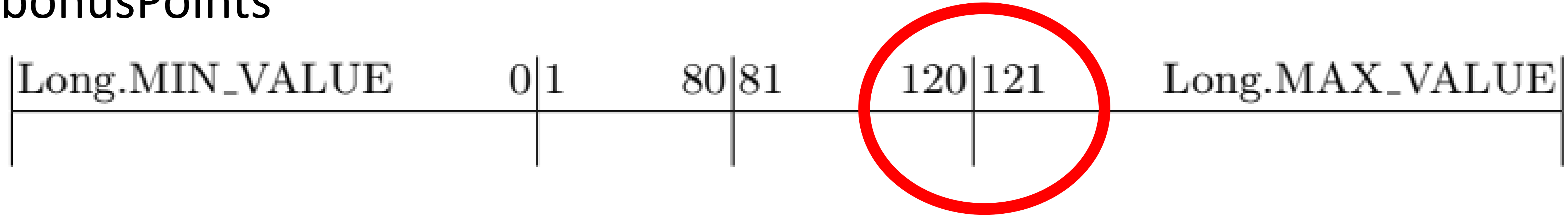
goldCustomer	true	false

- Return Value

Return Value	FULLPRICE	DISCOUNT	ERROR

Specification-Based Ranges

- bonusPoints



LIKELY FAULT LOCATION

Long.MIN_VALUE 0	1 80	81 120	121 Long.MAX_VALUE

- goldCustomer

true	false

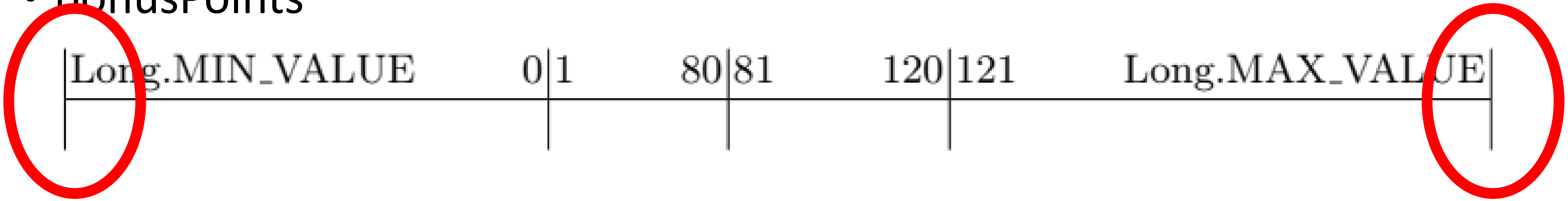
- Return Value

FULLPRICE	DISCOUNT	ERROR

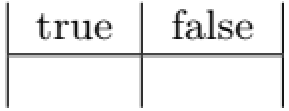
Specification-Based Ranges

LIKELY FAULT LOCATIONS

- bonusPoints

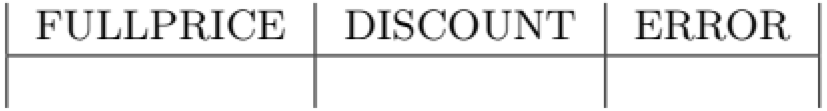


- goldCustomer

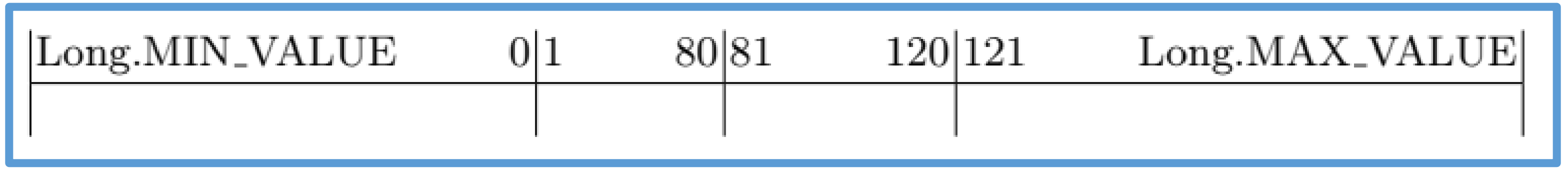


true	false

- Return Value

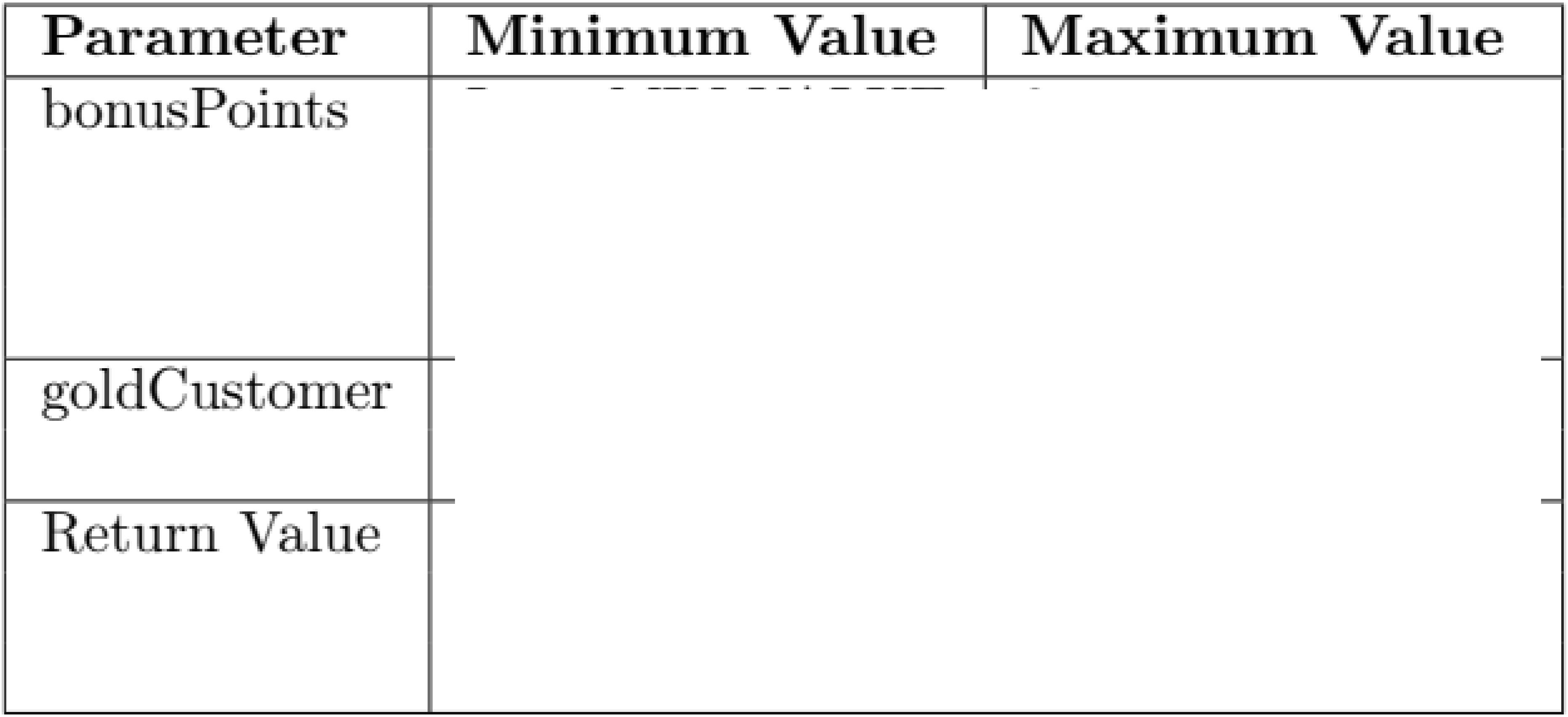


FULLPRICE	DISCOUNT	ERROR

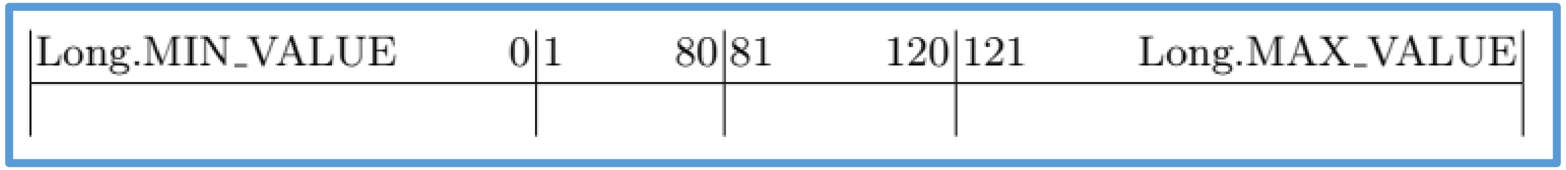


|Long.MIN_VALUE 0|1 80|81 120|121 Long.MAX_VALUE|

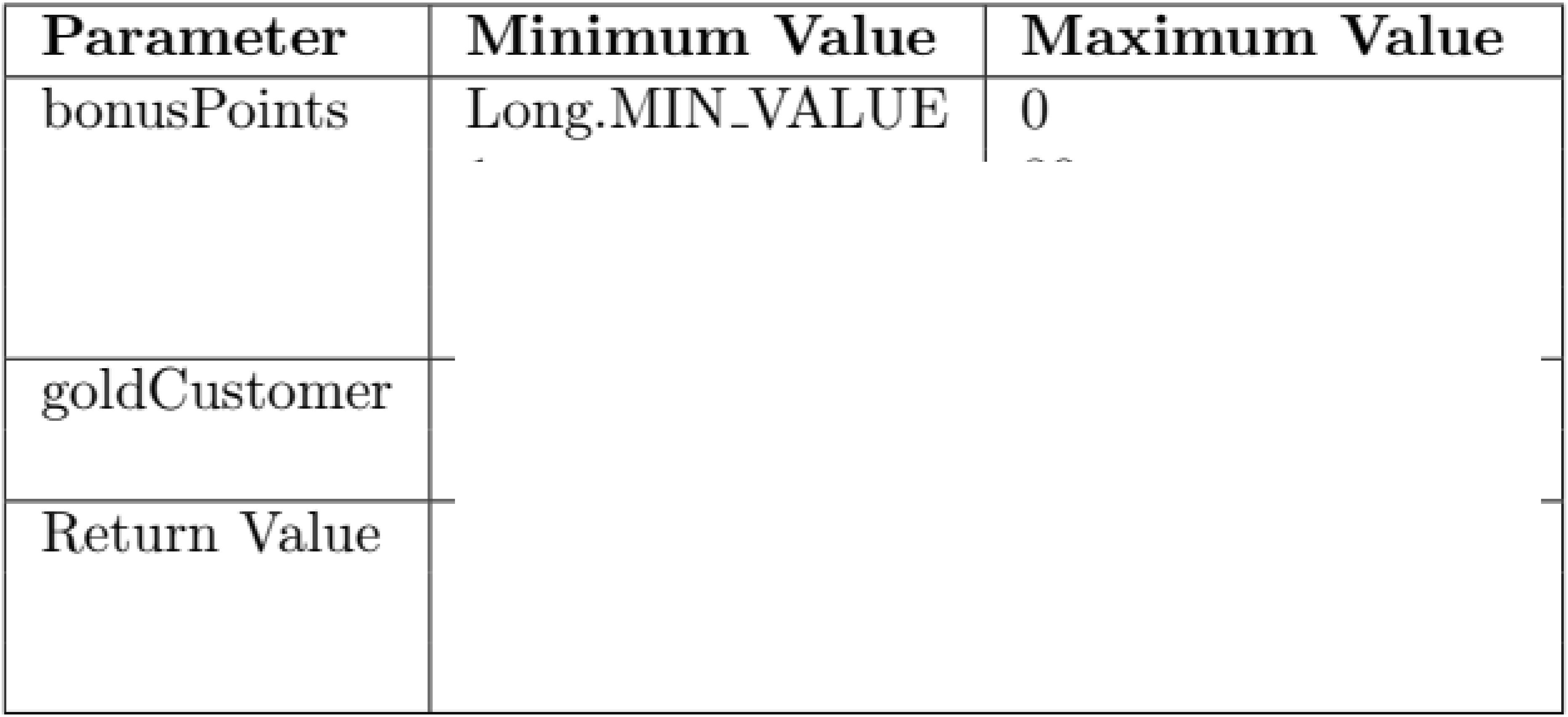
Boundary Values for giveDiscount()



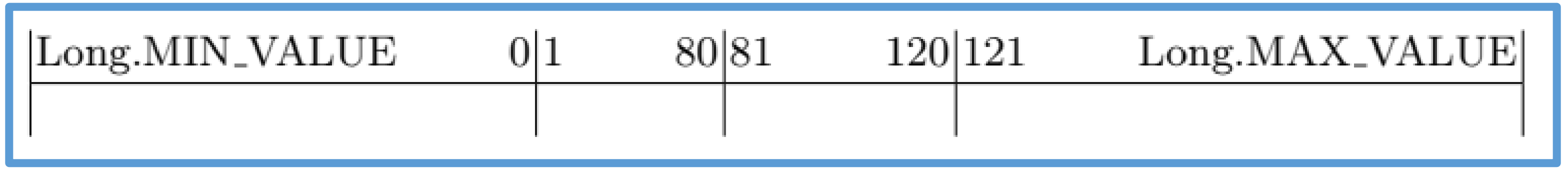
Parameter	Minimum Value	Maximum Value
bonusPoints		
goldCustomer		
Return Value		



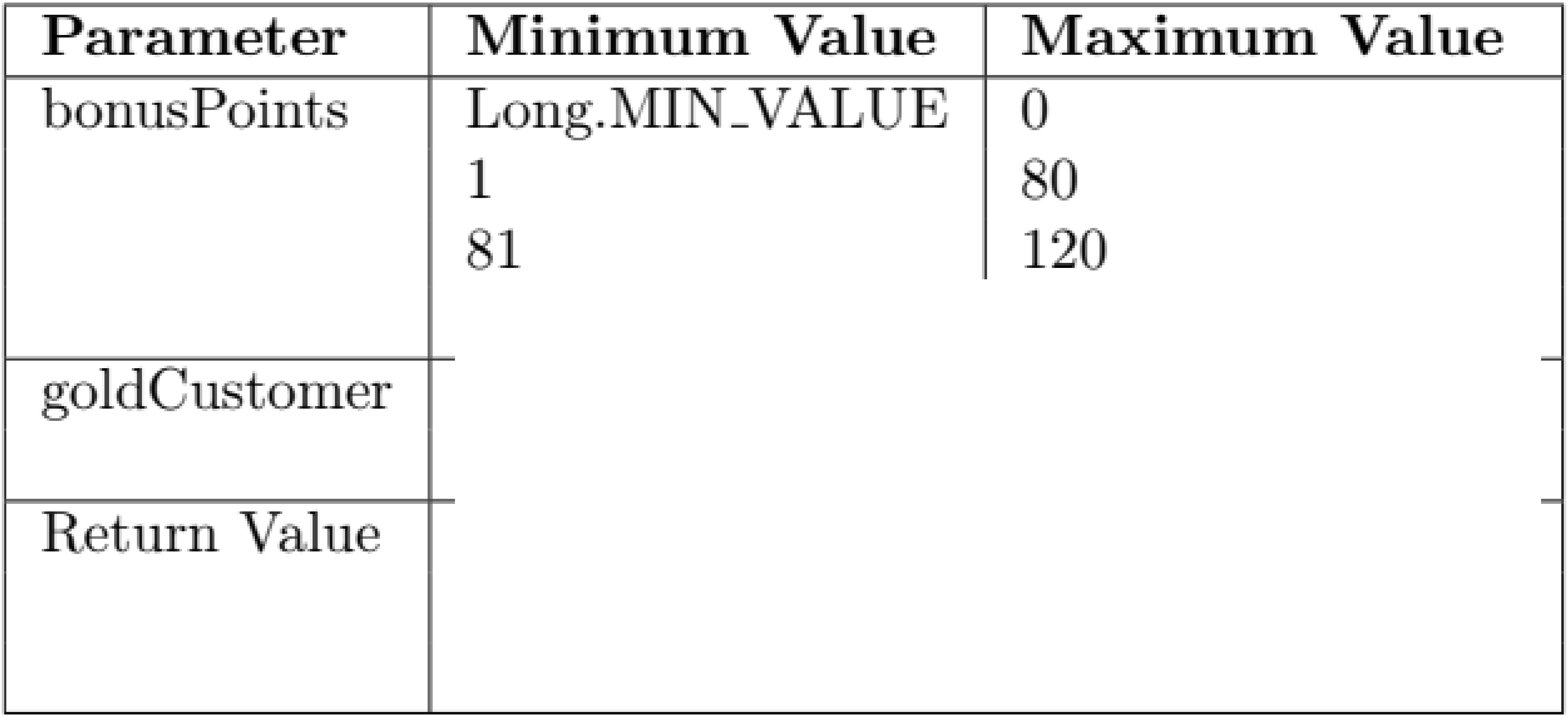
Boundary Values for giveDiscount()



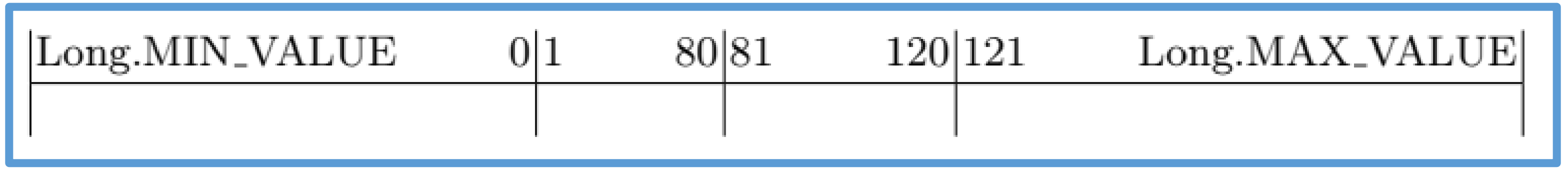
Parameter	Minimum Value	Maximum Value
bonusPoints	Long.MIN_VALUE	0
goldCustomer		
Return Value		



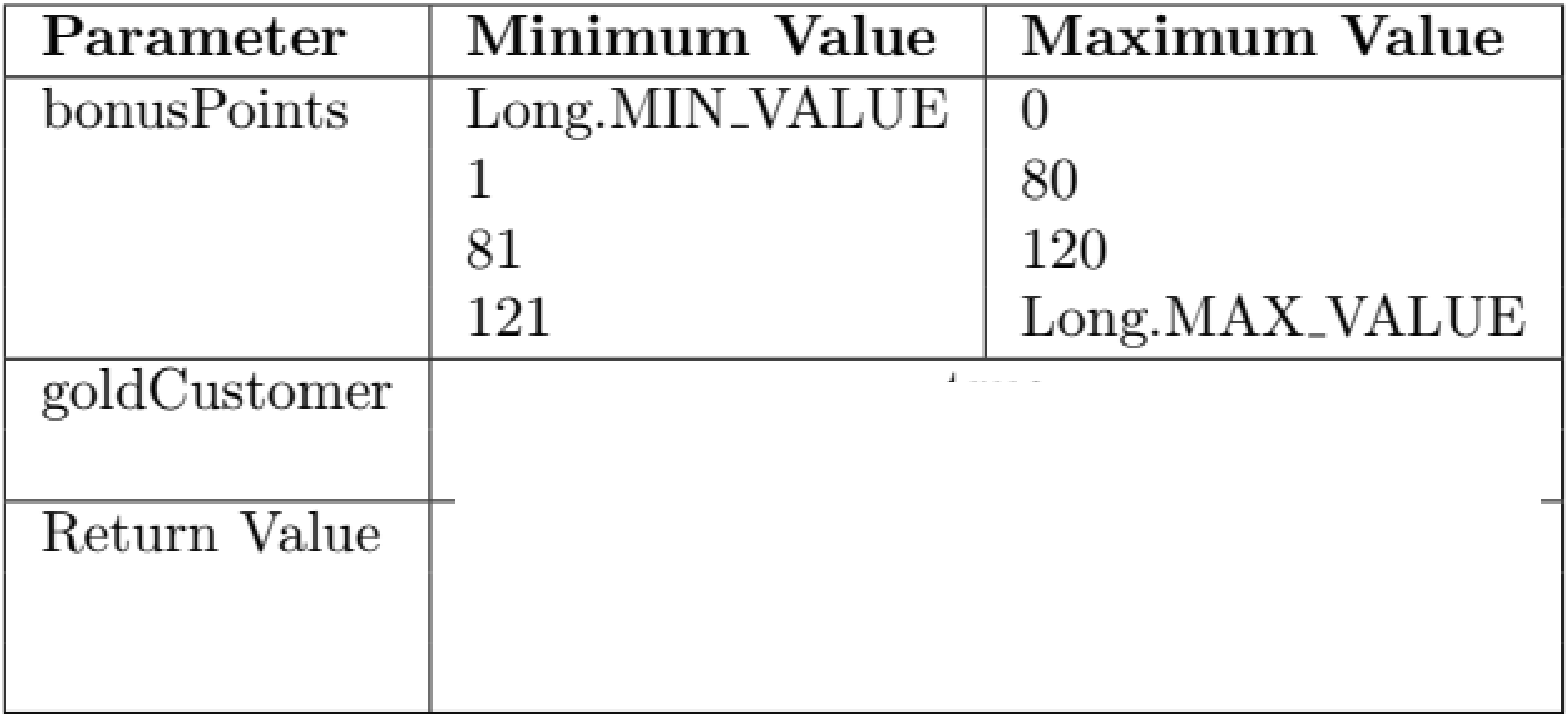
Boundary Values for giveDiscount()



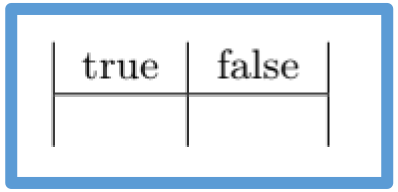
Parameter	Minimum Value	Maximum Value
bonusPoints	Long.MIN_VALUE	0
	1	80
	81	120
goldCustomer		
Return Value		



Boundary Values for giveDiscount()

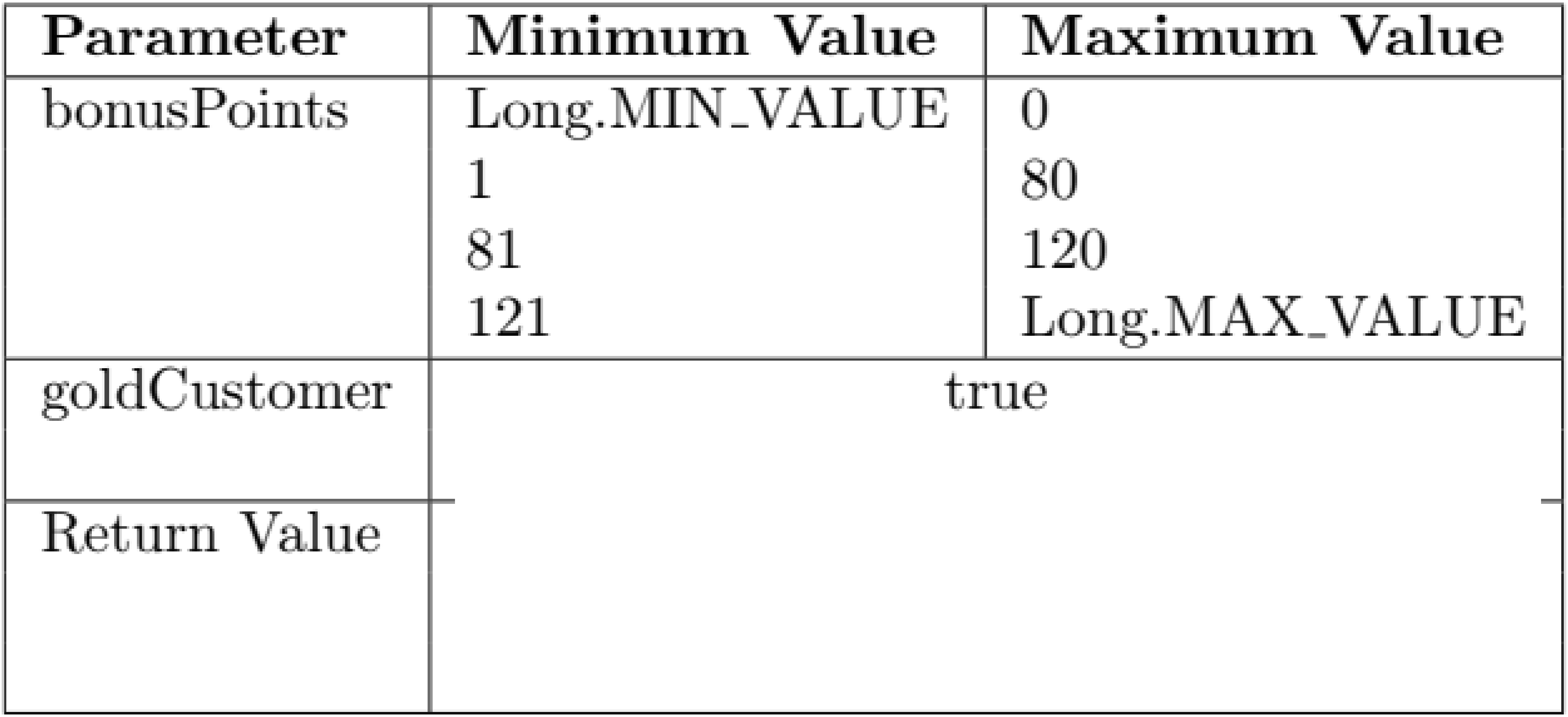


Parameter	Minimum Value	Maximum Value
bonusPoints	Long.MIN_VALUE	0
	1	80
	81	120
	121	Long.MAX_VALUE
goldCustomer		
Return Value		

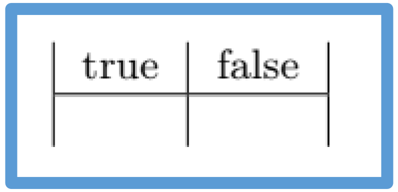


true	false

Boundary Values for giveDiscount()

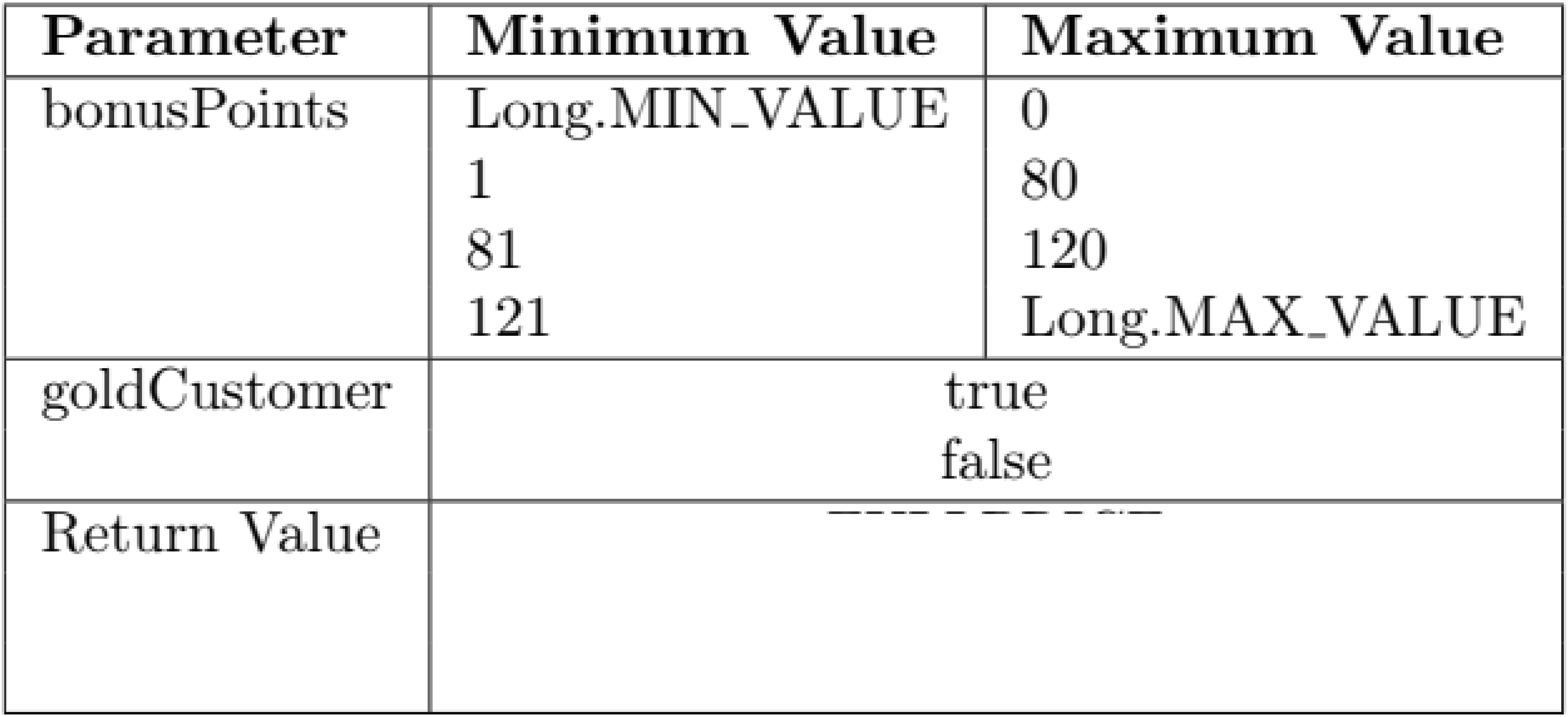


Parameter	Minimum Value	Maximum Value
bonusPoints	Long.MIN_VALUE 1 81 121	0 80 120 Long.MAX_VALUE
goldCustomer	true	
Return Value		

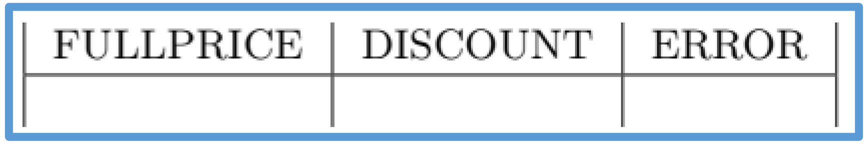


true	false

Boundary Values for giveDiscount()

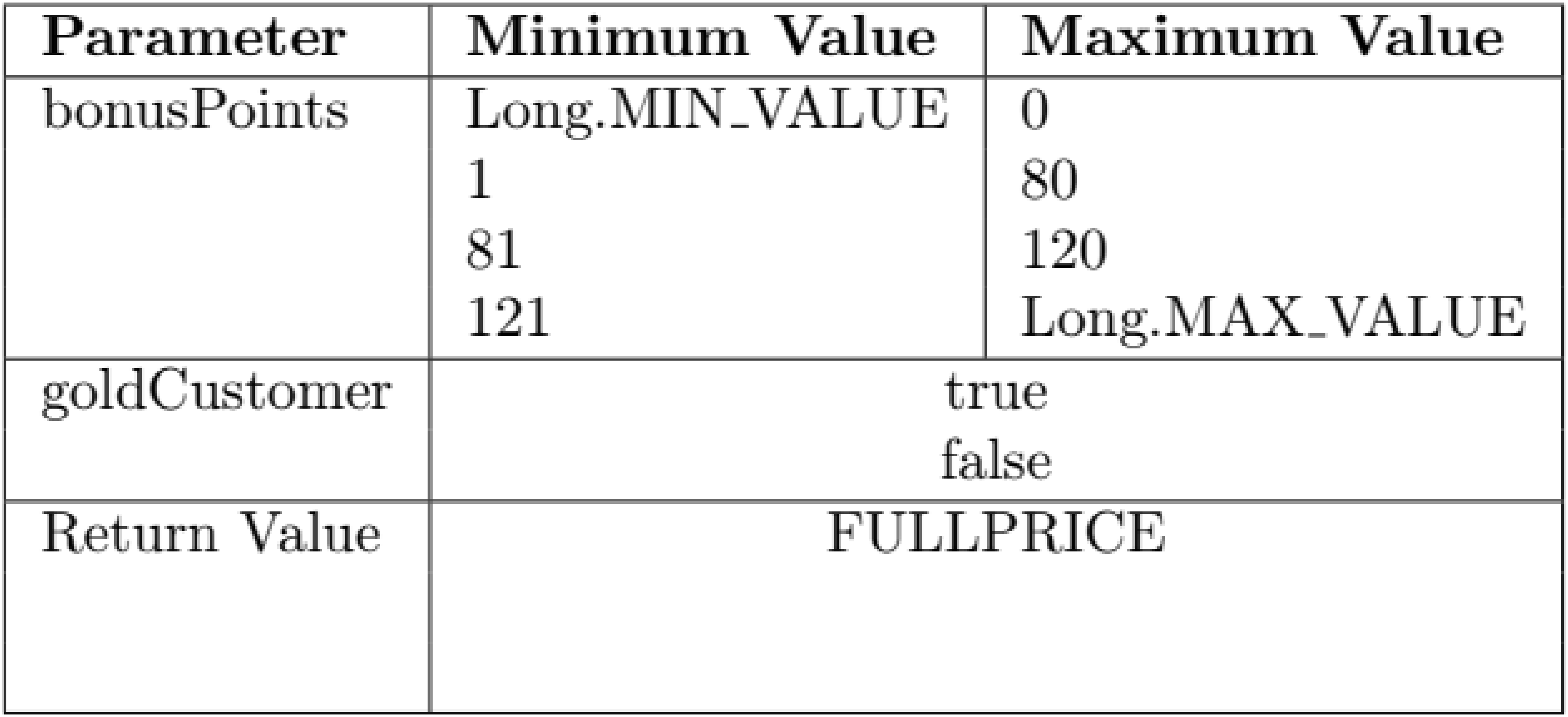


Parameter	Minimum Value	Maximum Value
bonusPoints	Long.MIN_VALUE	0
	1	80
	81	120
	121	Long.MAX_VALUE
goldCustomer	true false	
Return Value		

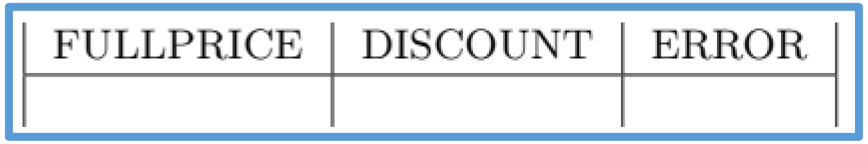


FULLPRICE	DISCOUNT	ERROR

Boundary Values for giveDiscount()

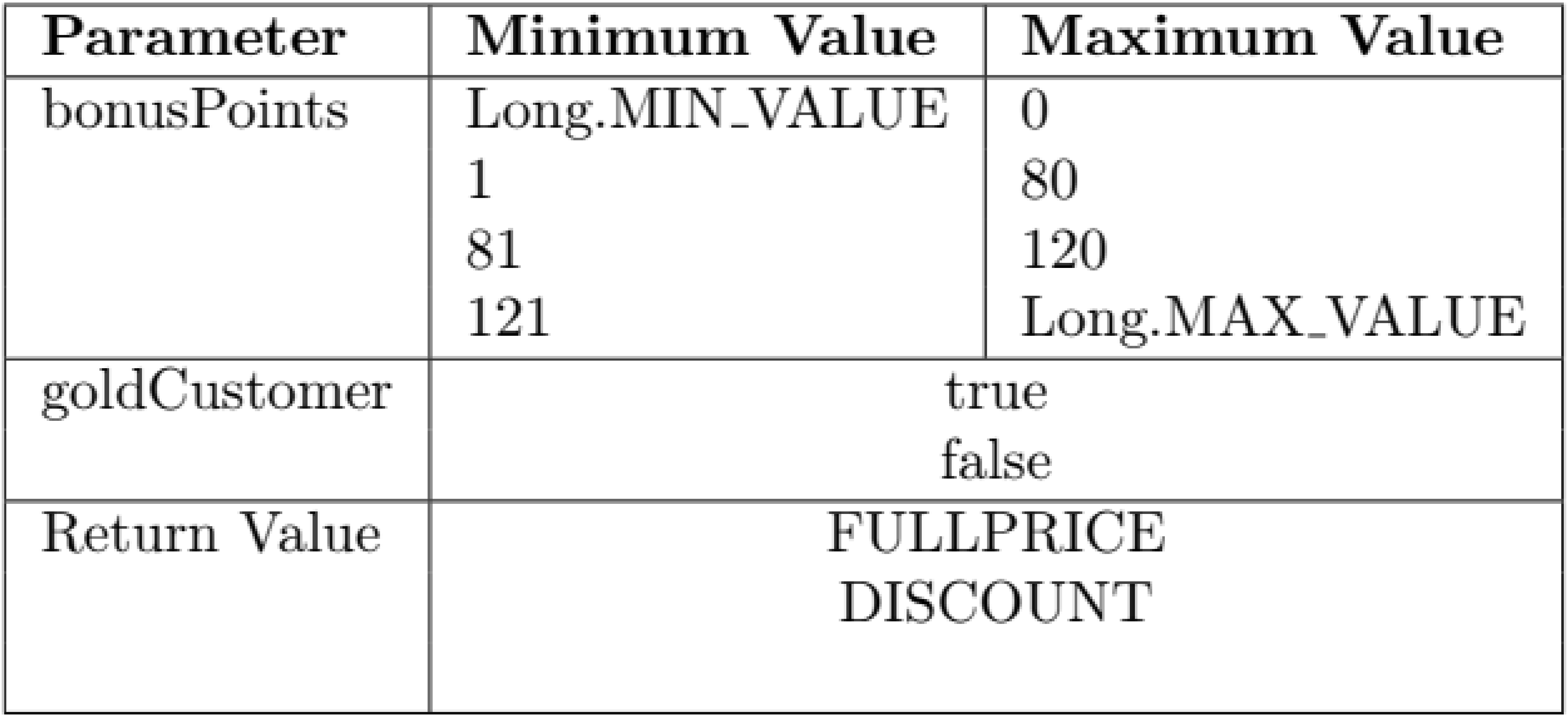


Parameter	Minimum Value	Maximum Value
bonusPoints	Long.MIN_VALUE	0
	1	80
	81	120
	121	Long.MAX_VALUE
goldCustomer	true	
	false	
Return Value	FULLPRICE	

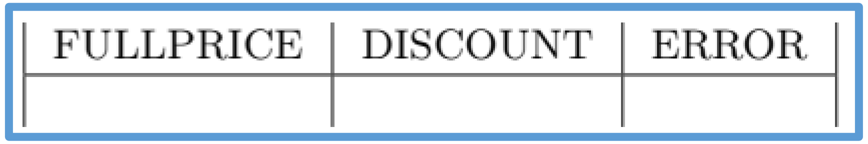


FULLPRICE	DISCOUNT	ERROR

Boundary Values for giveDiscount()

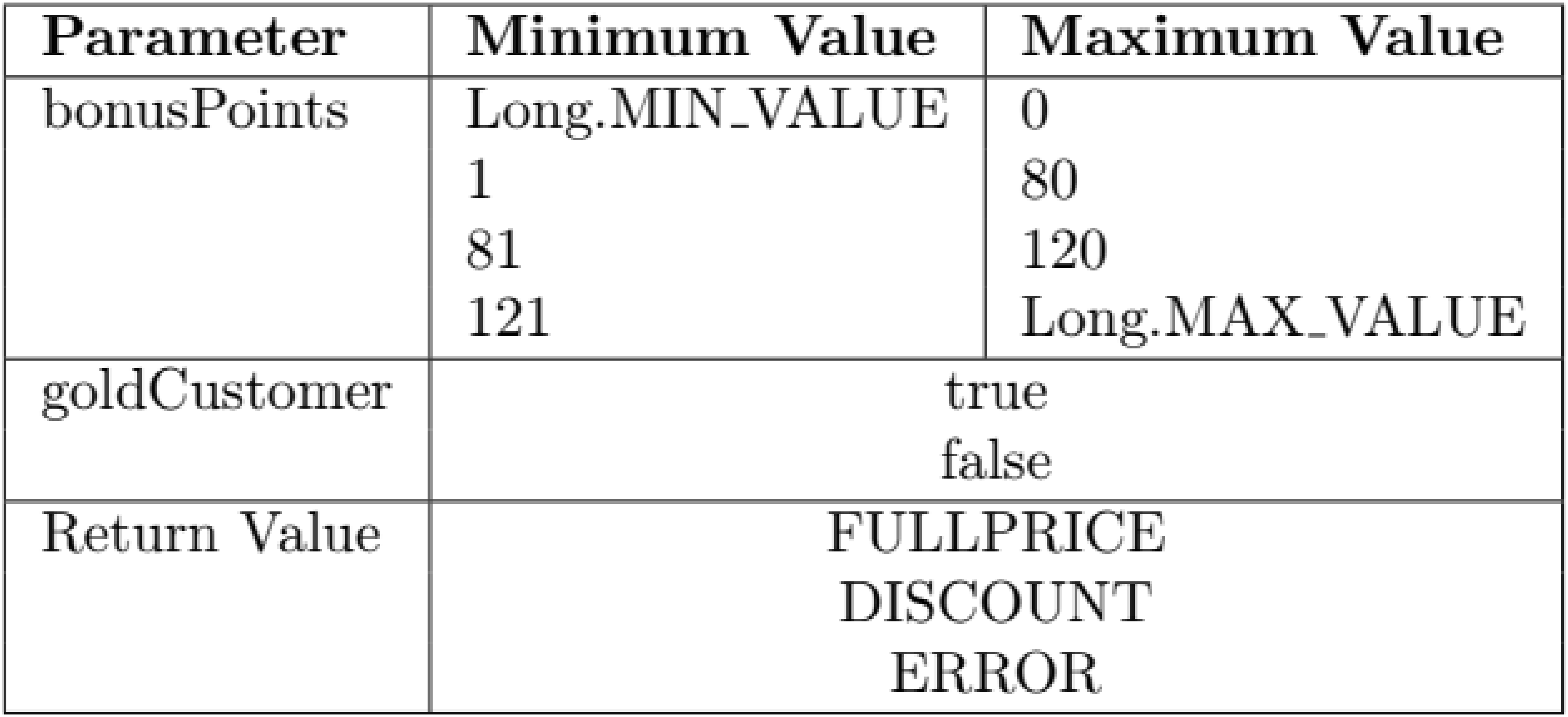


Parameter	Minimum Value	Maximum Value
bonusPoints	Long.MIN_VALUE 1 81 121	0 80 120 Long.MAX_VALUE
goldCustomer	true false	
Return Value	FULLPRICE DISCOUNT	



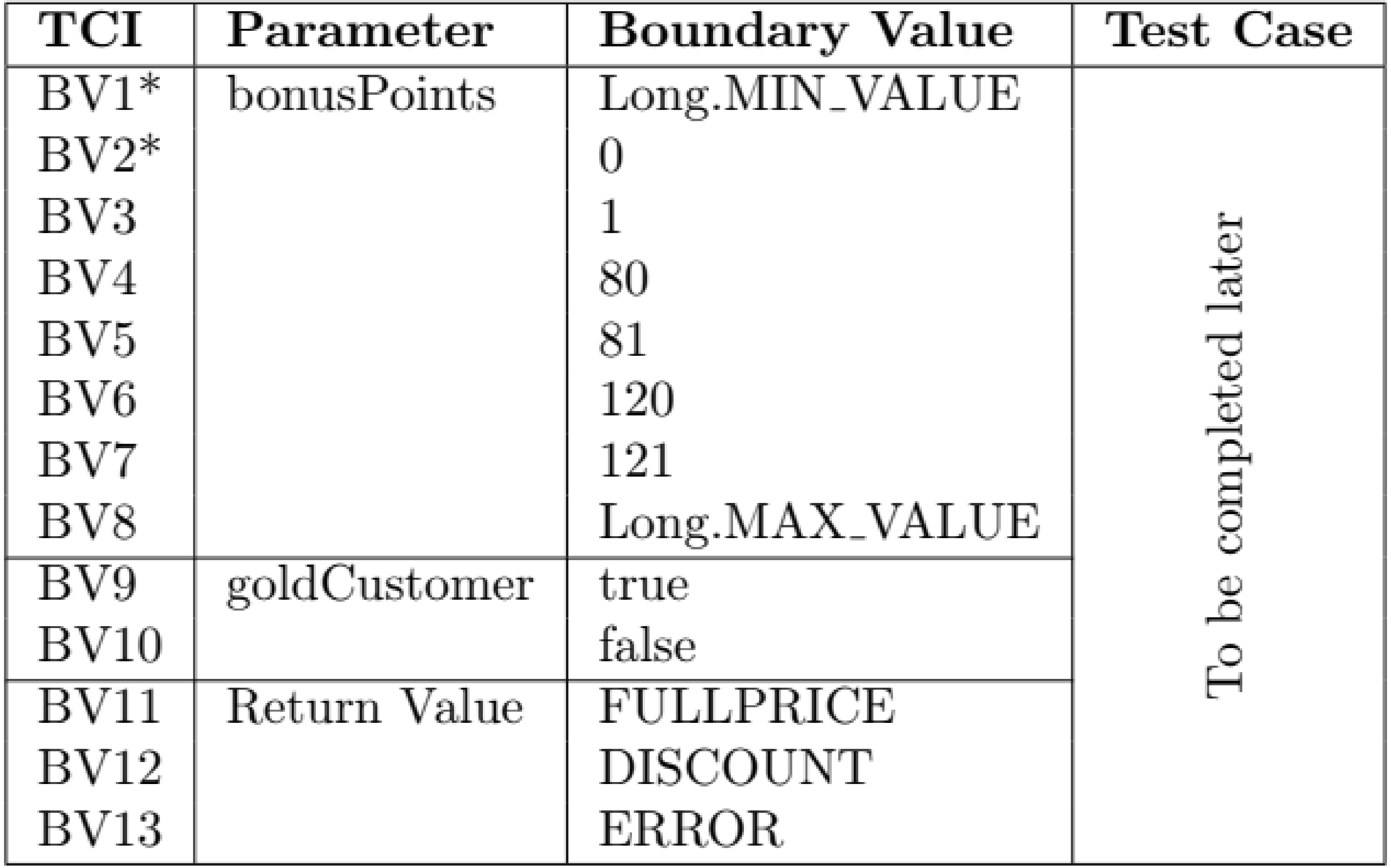
FULLPRICE	DISCOUNT	ERROR

Boundary Values for giveDiscount()



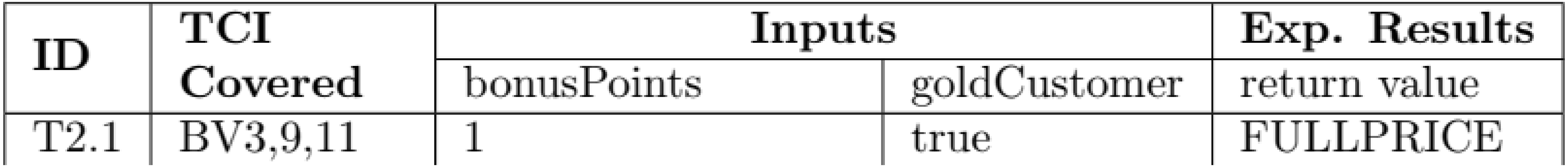
Parameter	**Minimum Value**	**Maximum Value**
bonusPoints	Long.MIN_VALUE	0
	1	80
	81	120
	121	Long.MAX_VALUE
goldCustomer	true false	
Return Value	FULLPRICE DISCOUNT ERROR	

2. Test Coverage Items



TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	To be completed later
BV2*		0	
BV3		1	
BV4		80	
BV5		81	
BV6		120	
BV7		121	
BV8		Long.MAX_VALUE	
BV9	goldCustomer	true	
BV10		false	
BV11	Return Value	FULLPRICE	
BV12		DISCOUNT	
BV13		ERROR	

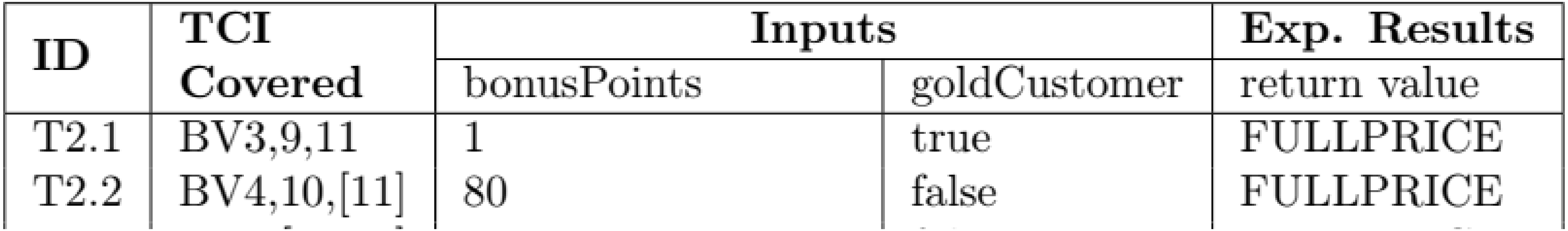
3. Test Cases (test 1)



ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE

- As for EP, cover the first non-error TCIs for each parameter
- BV3 is the first non-error TCI for bonusPoints
- BV9 is the first TCI for goldCustomer
- The specification states FULLPRICE for these inputs: BV11

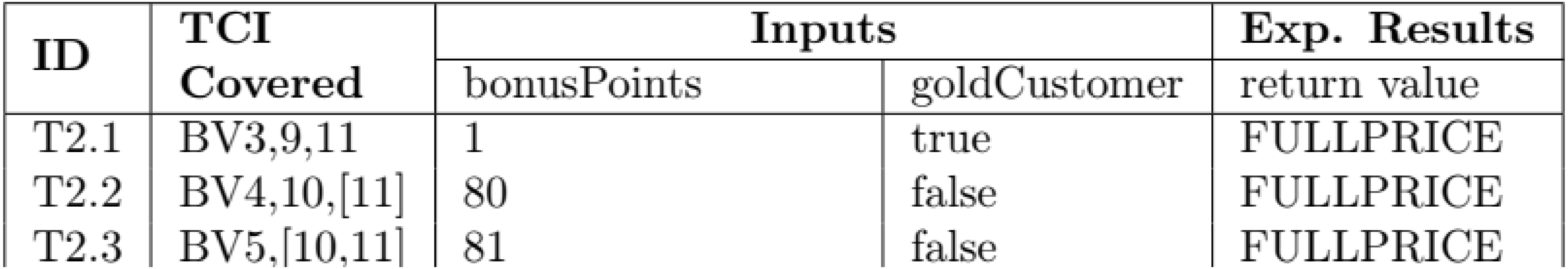
3. Test Cases (test 2)



ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE

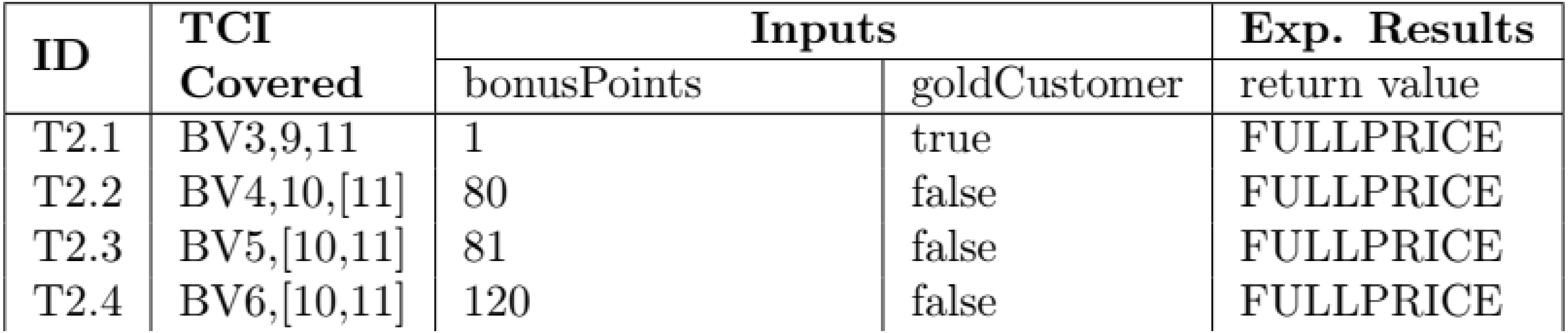
- BV4 is the next non-error TCI for bonusPoints
- BV10 is the next TCI for goldCustomer
- The specification states FULLPRICE for these inputs: BV11 (again, but unavoidable)

3. Test Cases (test 3)



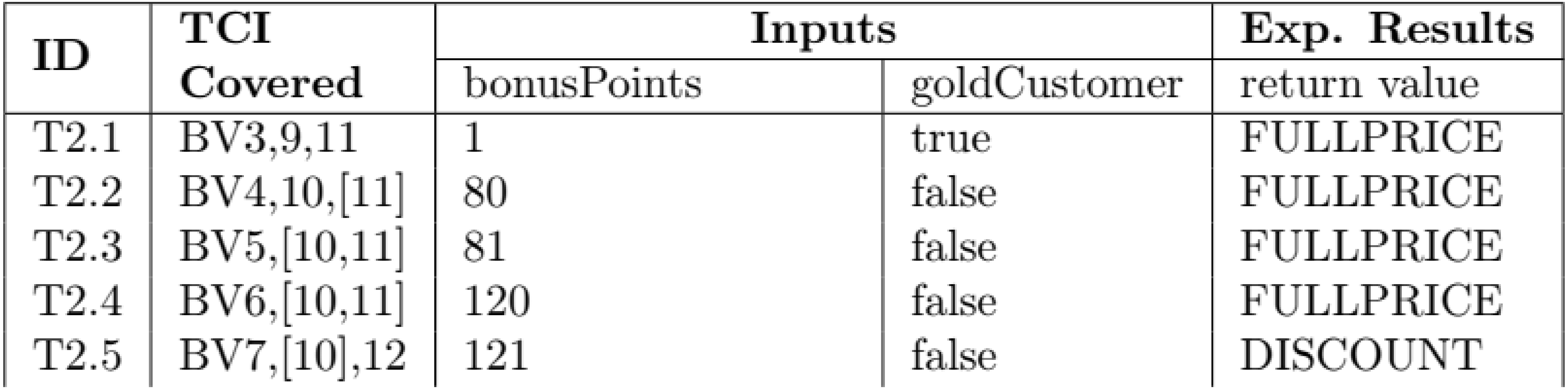
ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE

3. Test Cases (test 4)



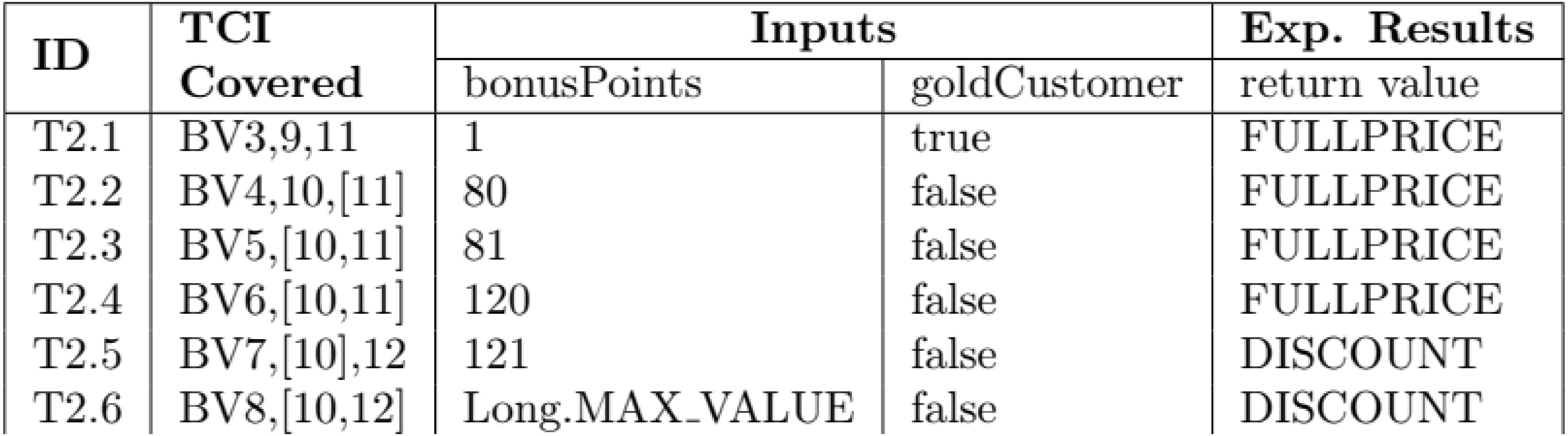
ID	**TCI Covered**	**Inputs**		**Exp. Results**
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE

3. Test Cases (test 5)



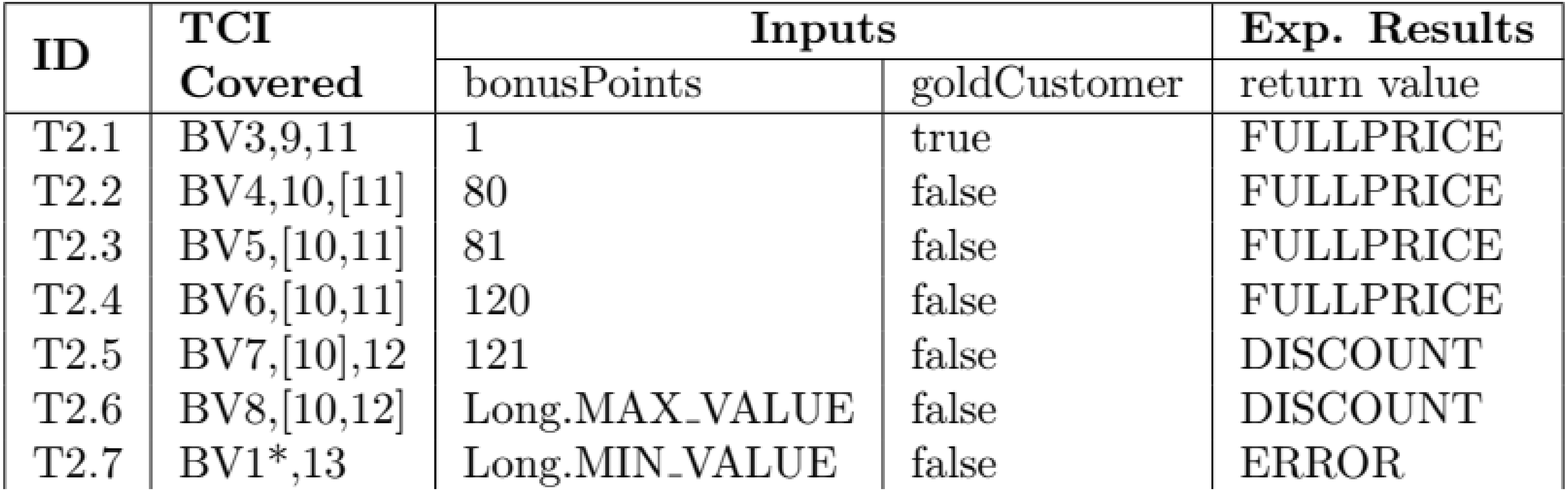
ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT

3. Test Cases (test 6)



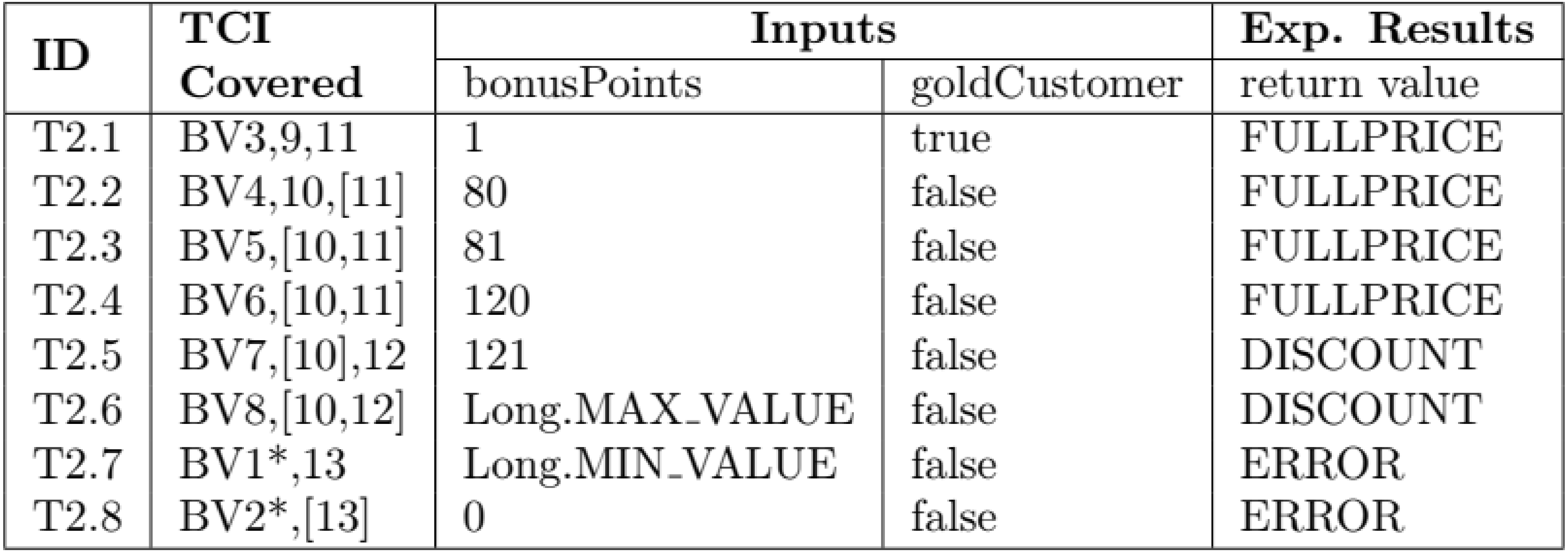
ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT

3. Test Cases (test 7)



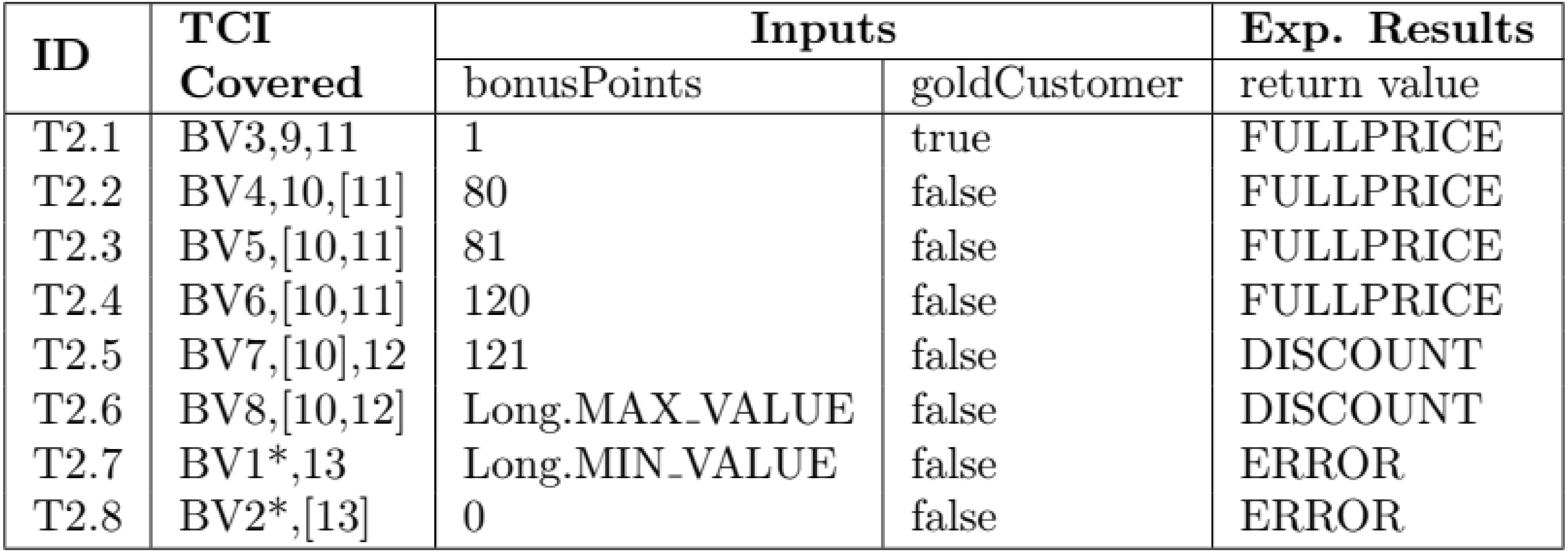
ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR

3. Test Cases (test 8)



ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

3. Test Cases (complete)



ID	TCI	Inputs		Exp. Results
	Covered	bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

Comment on BVA Test Cases

- Boundary value analysis appears to cover all the equivalence partition test coverage items – but not with typical values
- As for equivalence partitions, minimising the number of test cases can be an iterative process
- The target number of test cases is the largest number of boundary values for a parameter (here, it is 8 for bonusPoints)
- The technique is achieved if every test coverage item is covered

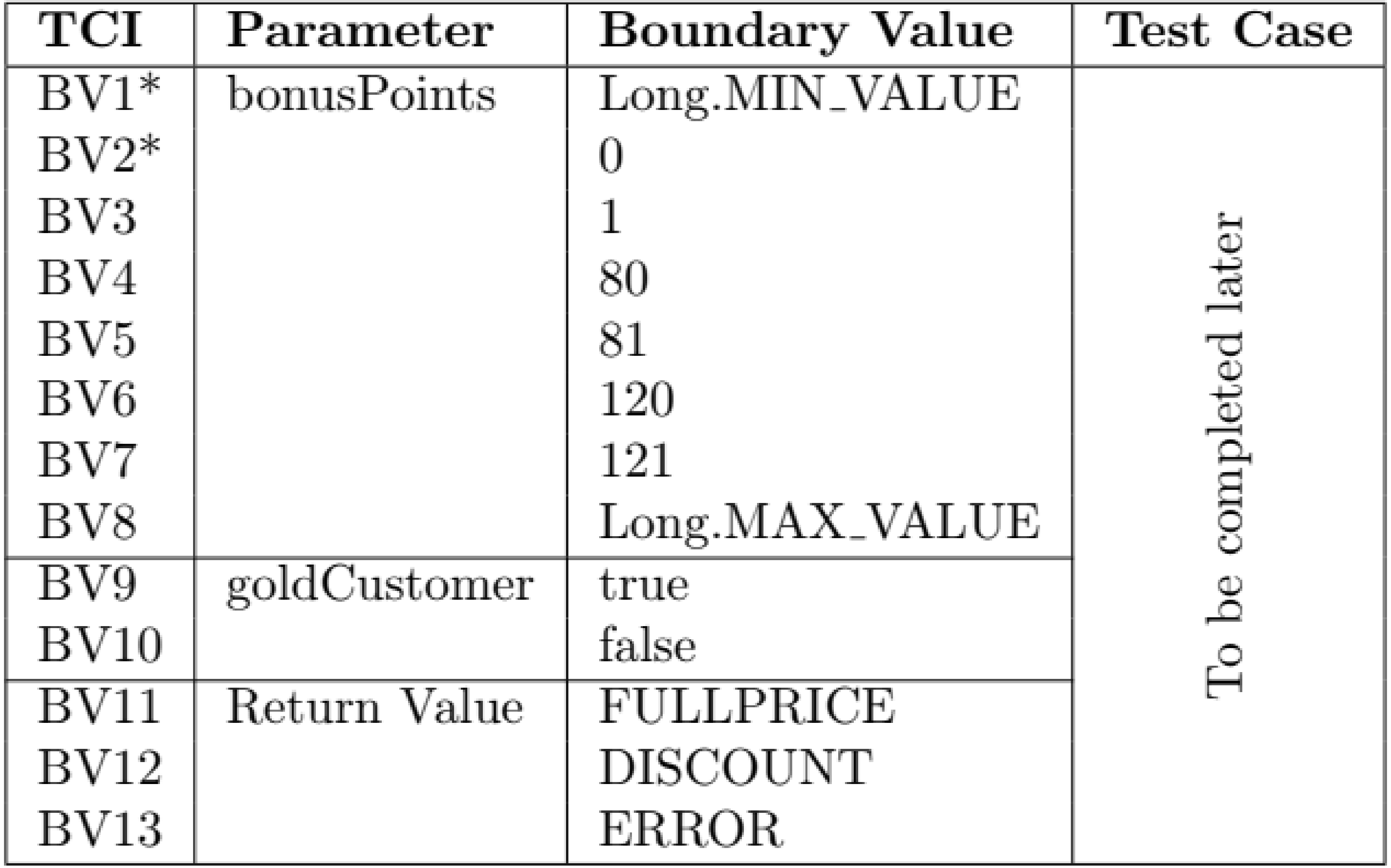
Comment on BVA Test Cases (continued)

- At the expense of approximately twice the number of tests, the minimum and maximum value of each equivalence partition has been tested at least once, using a minimum number of test cases
- Again, combinations of different values have not been exhaustively tested
- None of these test cases are duplicates of those developed for equivalence partitions, and so all of these require new test implementations

4. Test Design Verification

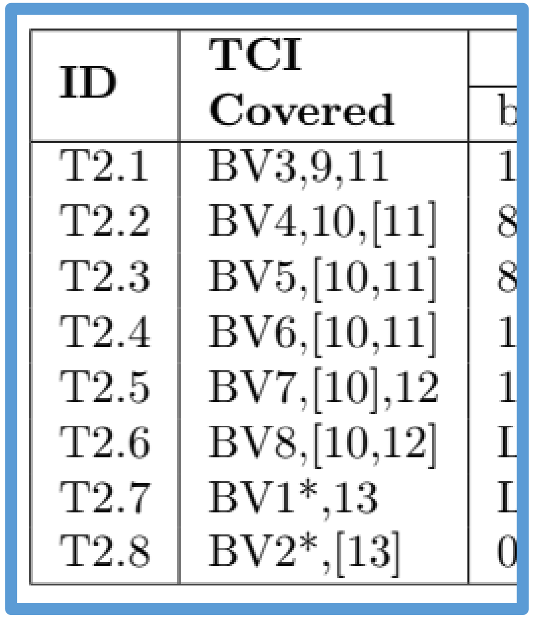
- Two parts:
  1. Complete the test coverage item table
  2. Review your work

Recap: TCI table

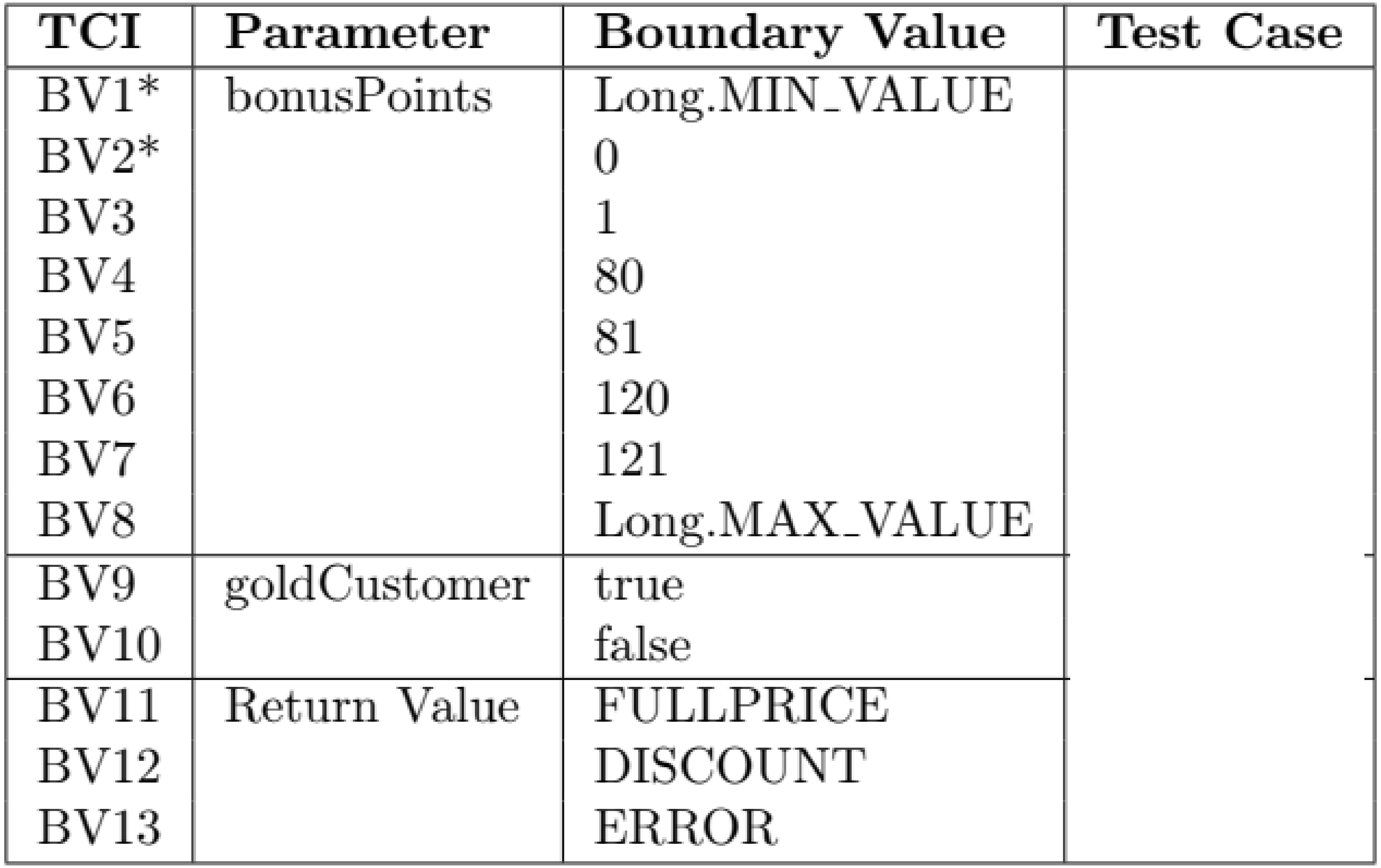


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	To be completed later
BV2*		0	
BV3		1	
BV4		80	
BV5		81	
BV6		120	
BV7		121	
BV8		Long.MAX_VALUE	
BV9	goldCustomer	true	
BV10		false	
BV11	Return Value	FULLPRICE	
BV12		DISCOUNT	
BV13		ERROR	

Completing the TCI table

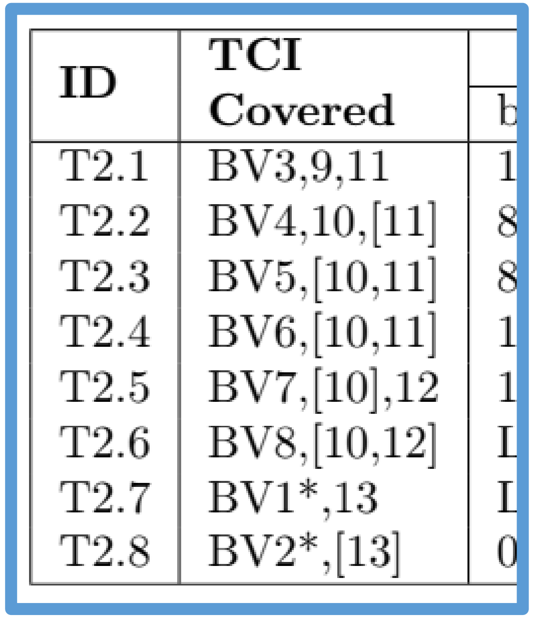


ID	TCI Covered	b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0

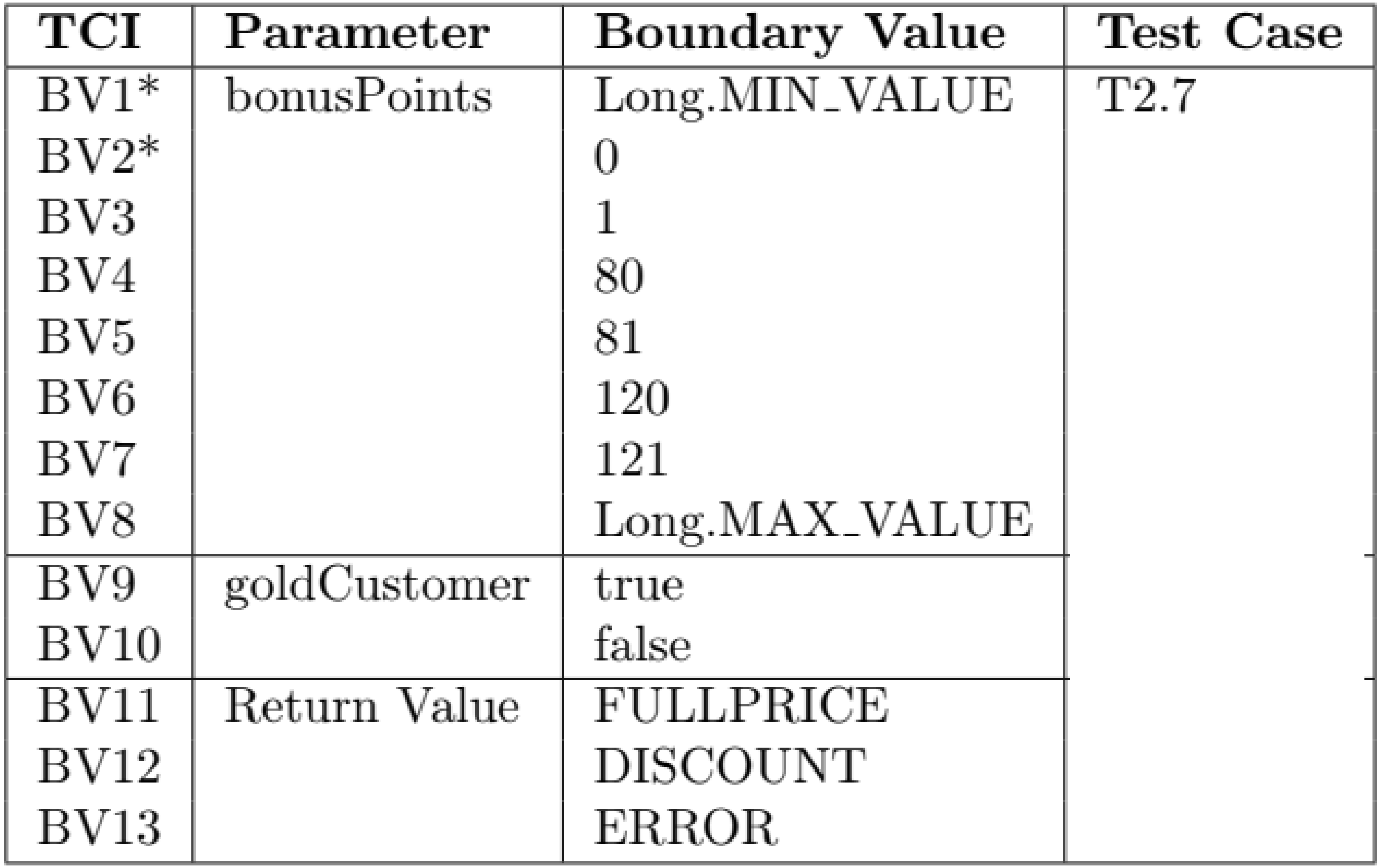


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	
BV2*		0	
BV3		1	
BV4		80	
BV5		81	
BV6		120	
BV7		121	
BV8		Long.MAX_VALUE	
BV9	goldCustomer	true	
BV10		false	
BV11	Return Value	FULLPRICE	
BV12		DISCOUNT	
BV13		ERROR	

Completing the TCI table

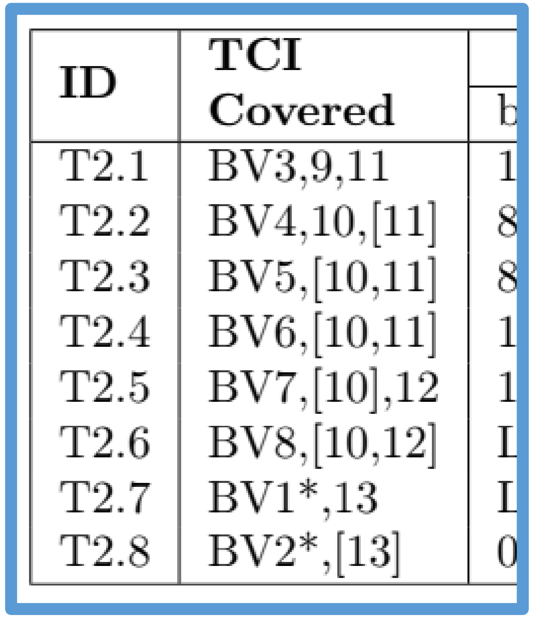


ID	TCI Covered	b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0

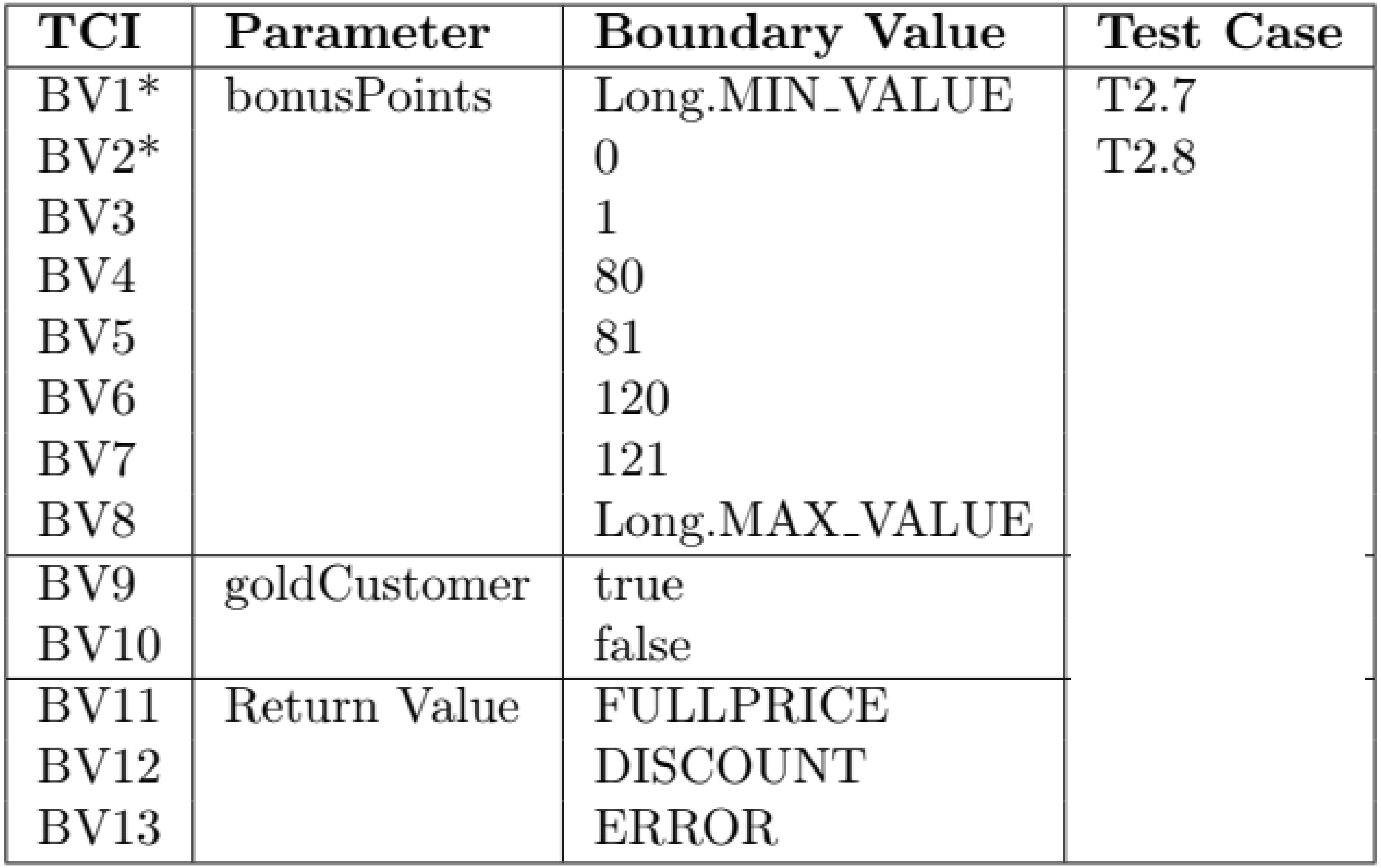


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	
BV3		1	
BV4		80	
BV5		81	
BV6		120	
BV7		121	
BV8		Long.MAX_VALUE	
BV9	goldCustomer	true	
BV10		false	
BV11	Return Value	FULLPRICE	
BV12		DISCOUNT	
BV13		ERROR	

Completing the TCI table

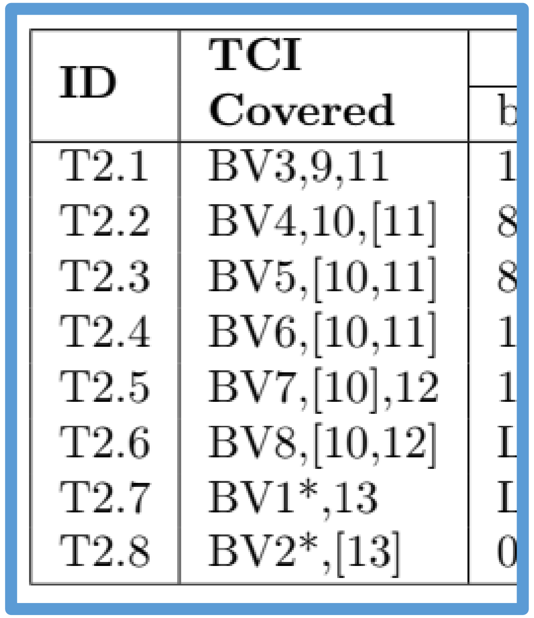


ID	TCI Covered	b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0

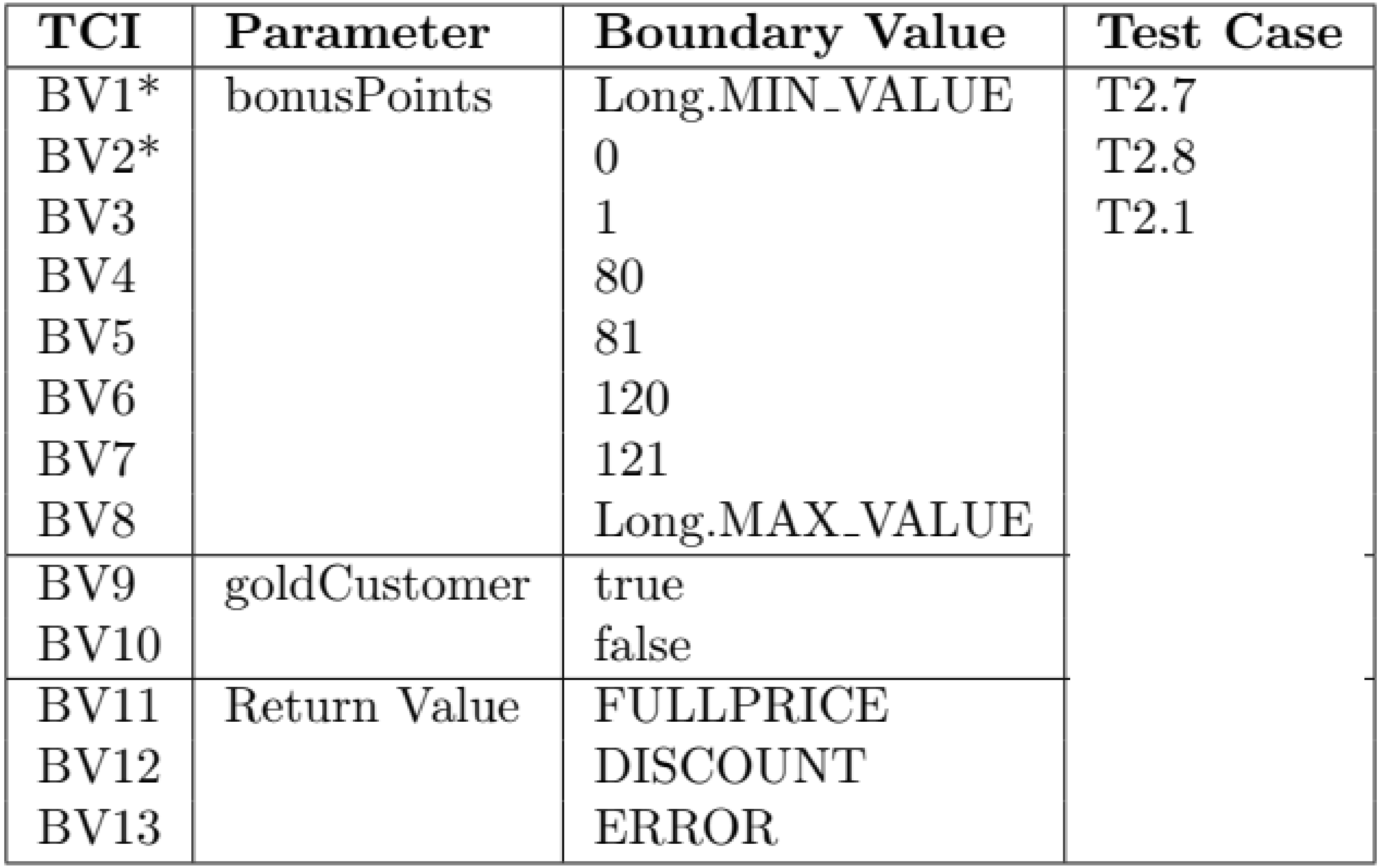


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	T2.8
BV3		1	
BV4		80	
BV5		81	
BV6		120	
BV7		121	
BV8		Long.MAX_VALUE	
BV9	goldCustomer	true	
BV10		false	
BV11	Return Value	FULLPRICE	
BV12		DISCOUNT	
BV13		ERROR	

Completing the TCI table

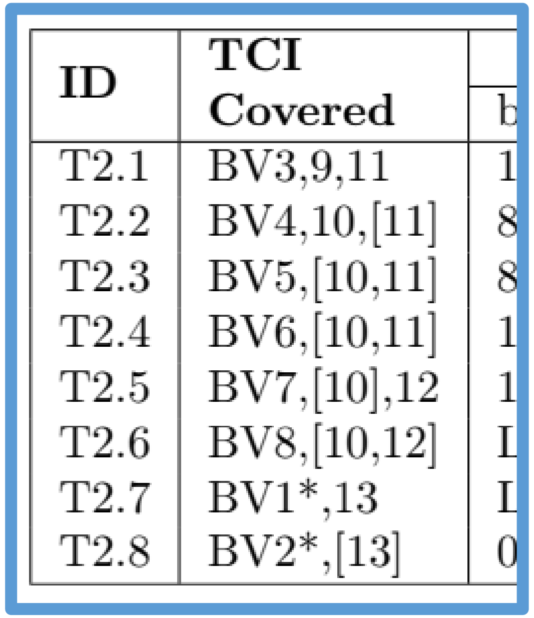


ID	TCI Covered	b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0

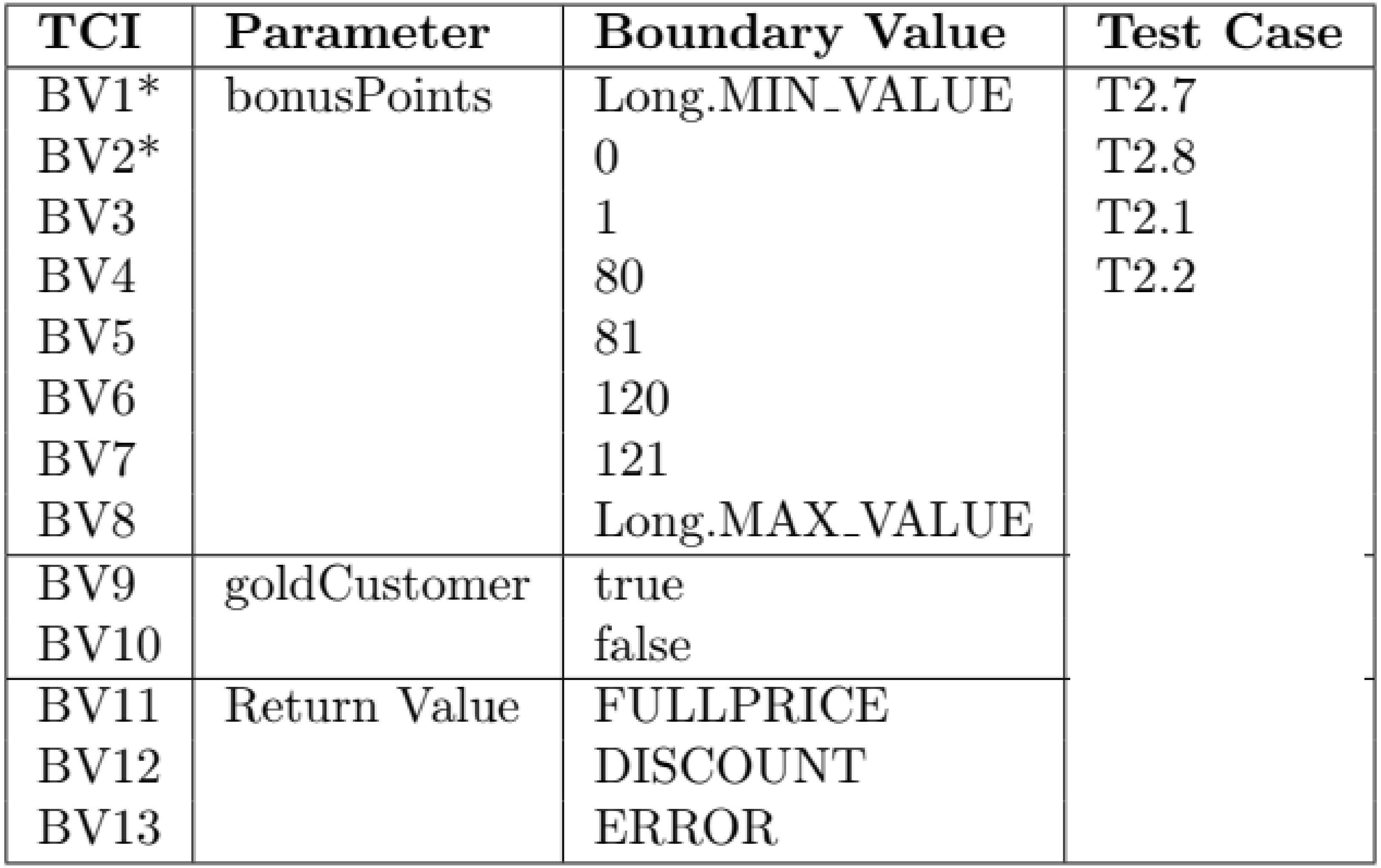


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	T2.8
BV3		1	T2.1
BV4		80	
BV5		81	
BV6		120	
BV7		121	
BV8		Long.MAX_VALUE	
BV9	goldCustomer	true	
BV10		false	
BV11	Return Value	FULLPRICE	
BV12		DISCOUNT	
BV13		ERROR	

Completing the TCI table

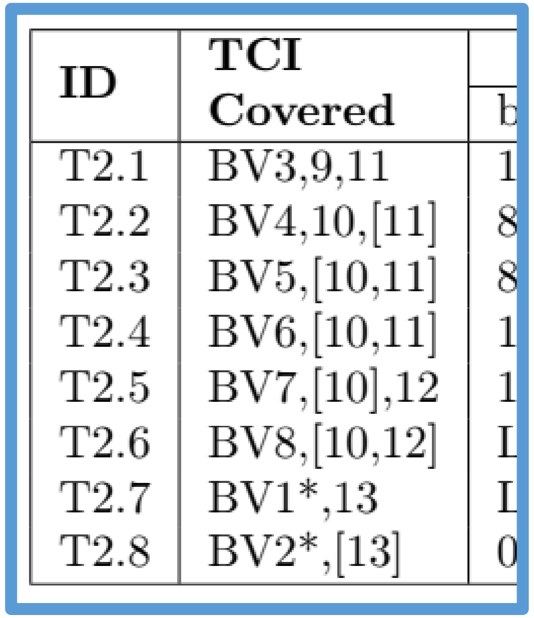


ID	TCI Covered	b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0

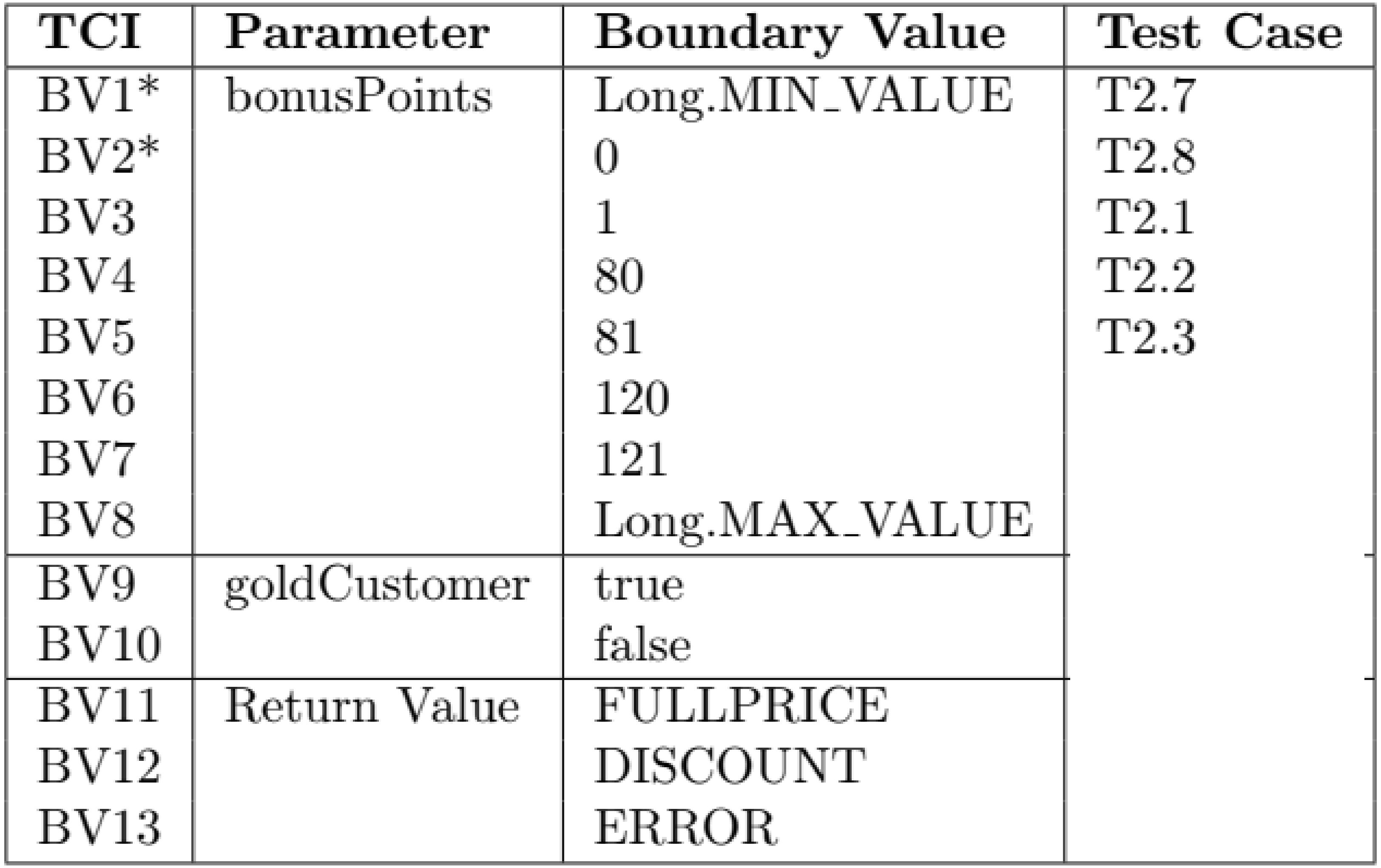


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	T2.8
BV3		1	T2.1
BV4		80	T2.2
BV5		81	
BV6		120	
BV7		121	
BV8		Long.MAX_VALUE	
BV9	goldCustomer	true	
BV10		false	
BV11	Return Value	FULLPRICE	
BV12		DISCOUNT	
BV13		ERROR	

Completing the TCI table

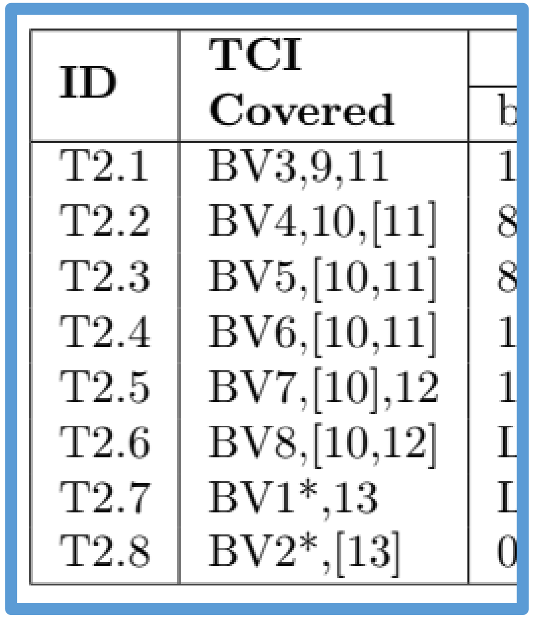


ID	TCI Covered	b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0

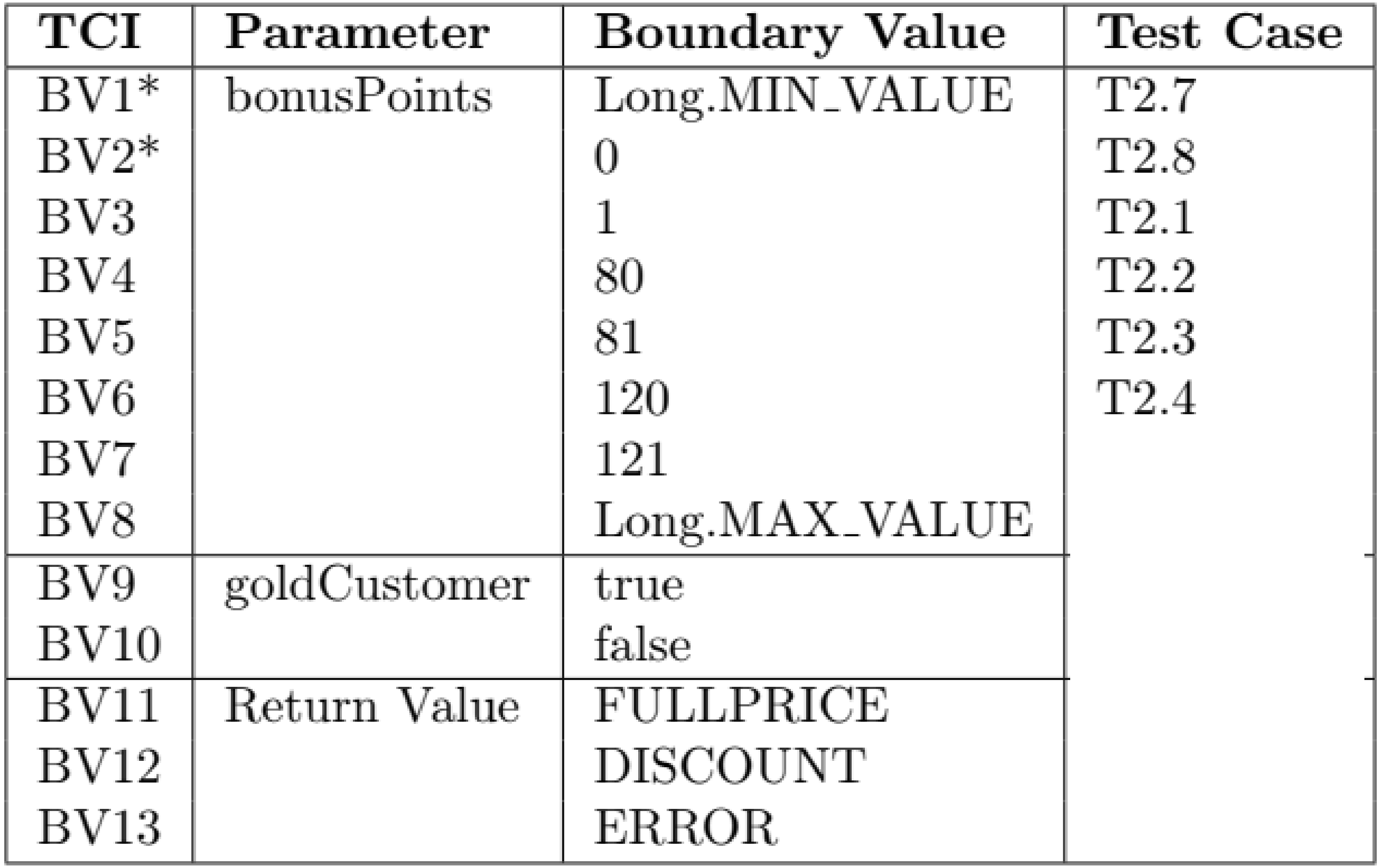


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	T2.8
BV3		1	T2.1
BV4		80	T2.2
BV5		81	T2.3
BV6		120	
BV7		121	
BV8		Long.MAX_VALUE	
BV9	goldCustomer	true	
BV10		false	
BV11	Return Value	FULLPRICE	
BV12		DISCOUNT	
BV13		ERROR	

Completing the TCI table

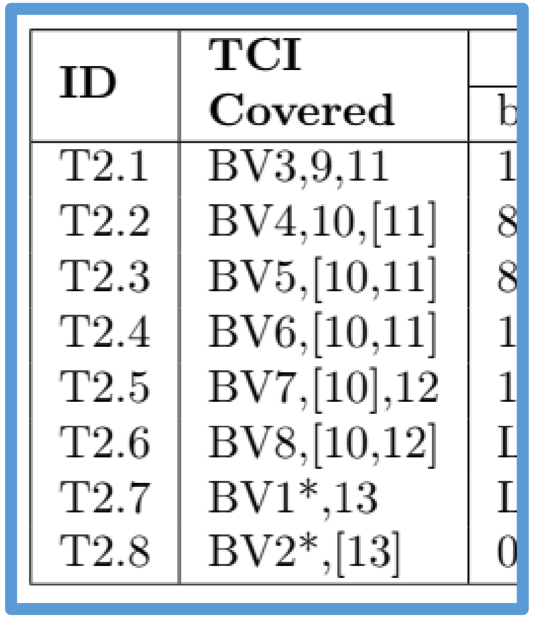


ID	TCI Covered	
		b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0

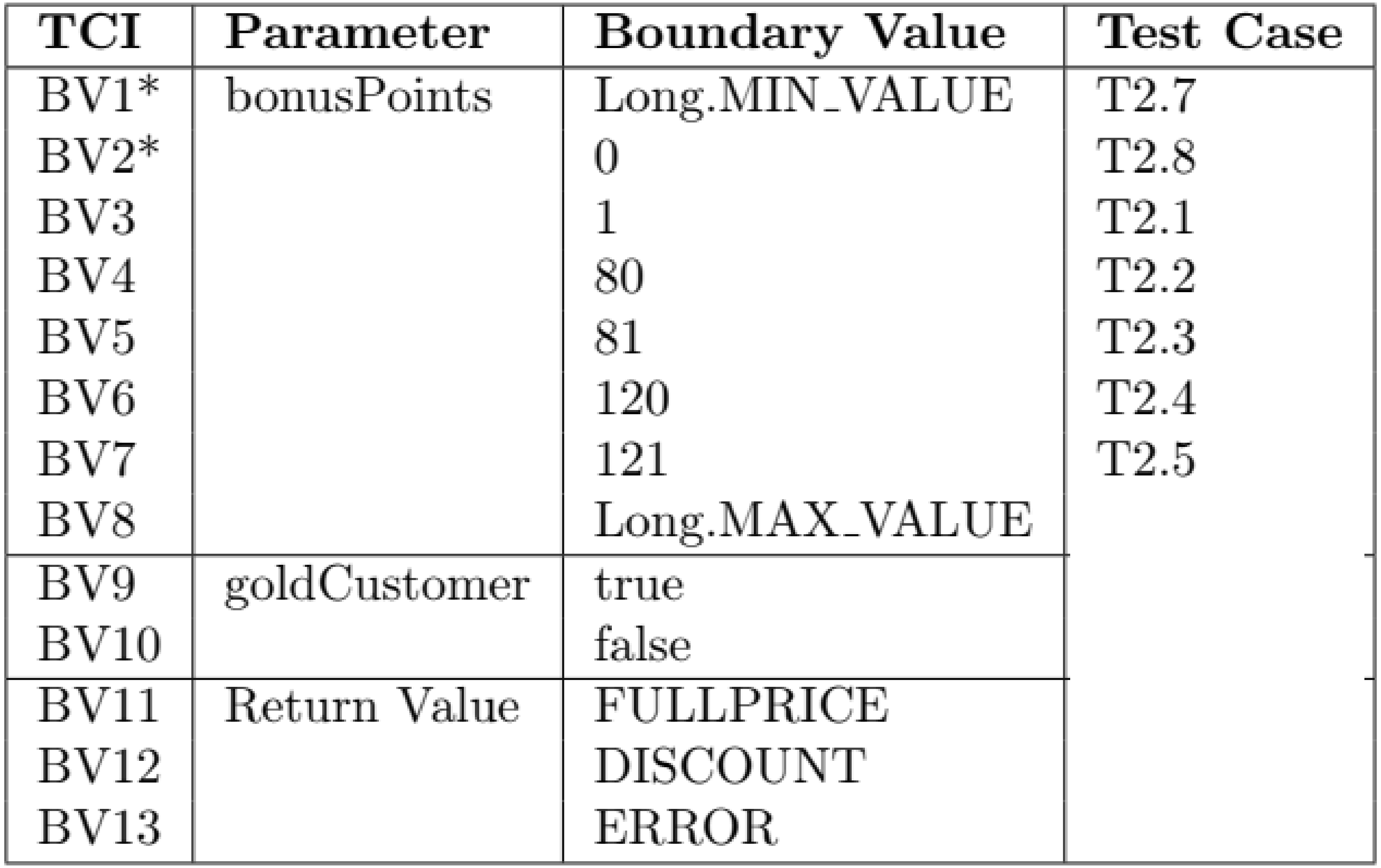


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	T2.8
BV3		1	T2.1
BV4		80	T2.2
BV5		81	T2.3
BV6		120	T2.4
BV7		121	
BV8		Long.MAX_VALUE	
BV9	goldCustomer	true	
BV10		false	
BV11	Return Value	FULLPRICE	
BV12		DISCOUNT	
BV13		ERROR	

Completing the TCI table

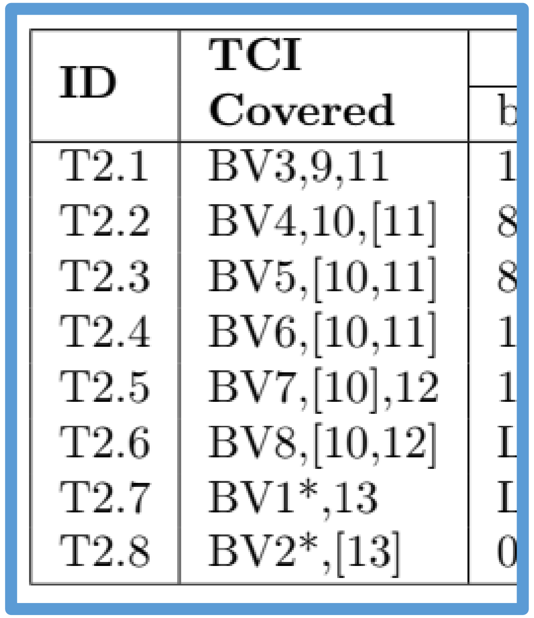


ID	TCI Covered	b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0

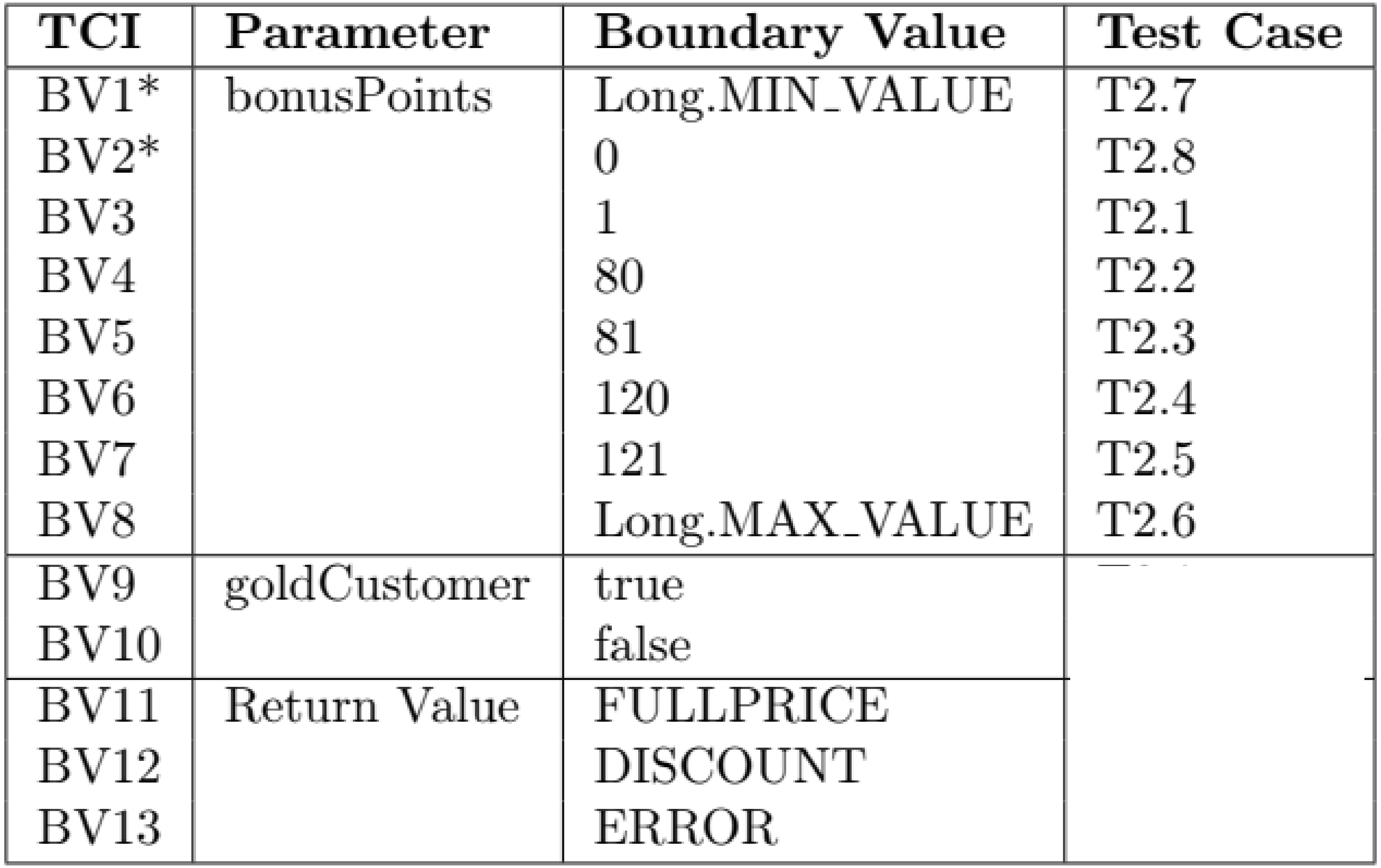


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	T2.8
BV3		1	T2.1
BV4		80	T2.2
BV5		81	T2.3
BV6		120	T2.4
BV7		121	T2.5
BV8		Long.MAX_VALUE	
BV9	goldCustomer	true	
BV10		false	
BV11	Return Value	FULLPRICE	
BV12		DISCOUNT	
BV13		ERROR	

Completing the TCI table

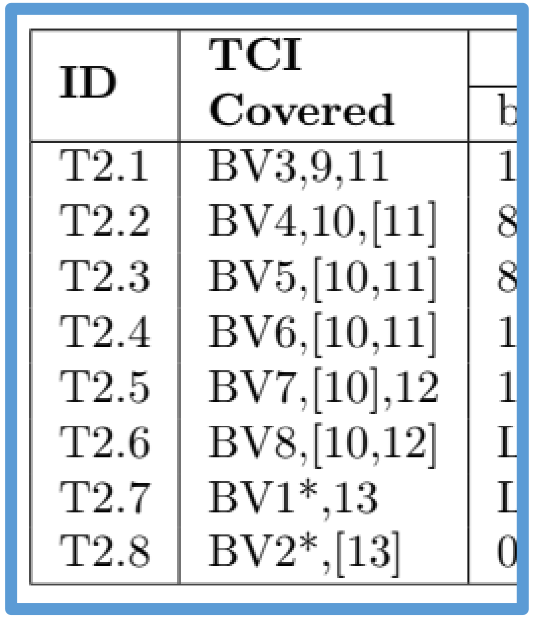


ID	TCI Covered	b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0

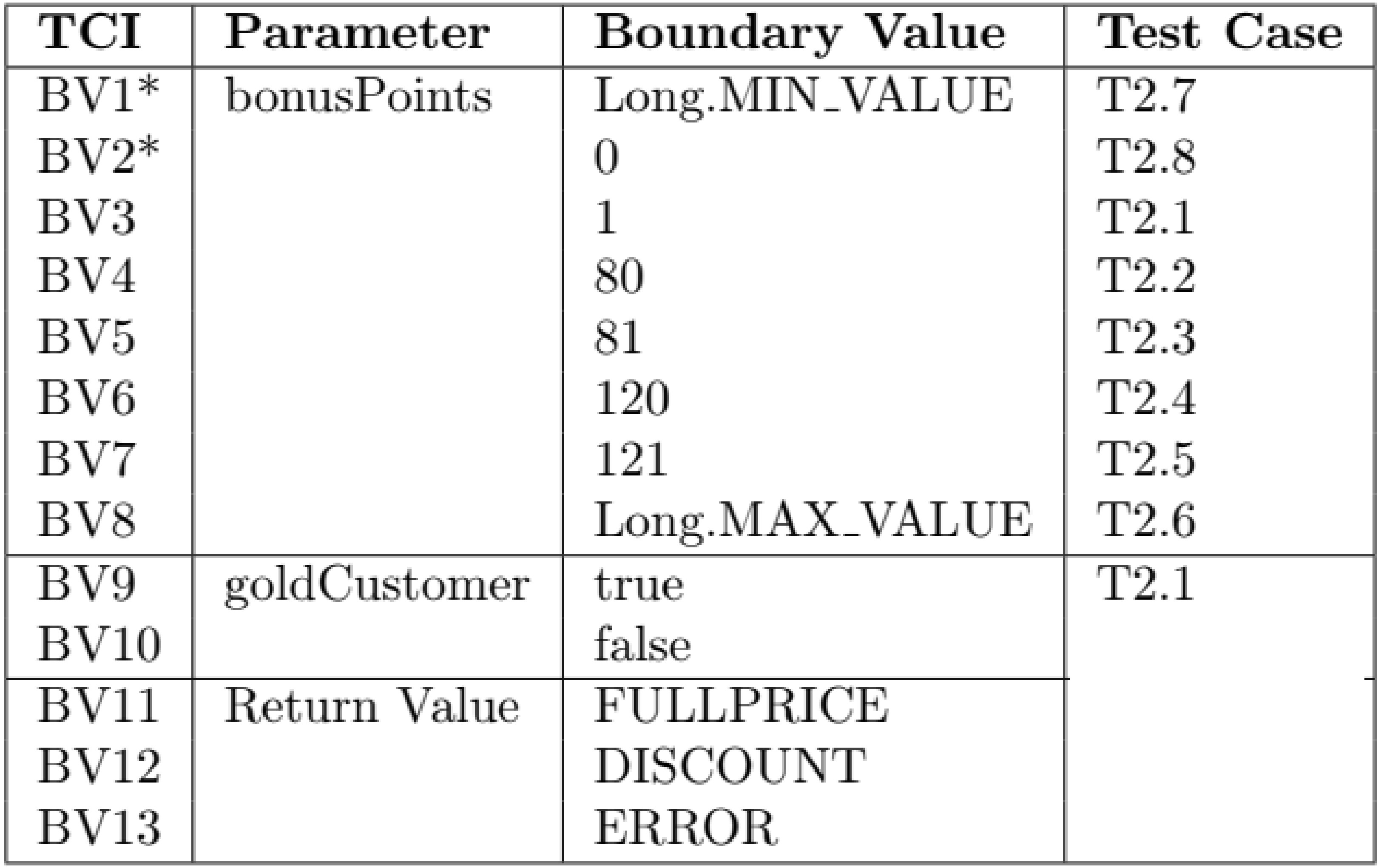


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	T2.8
BV3		1	T2.1
BV4		80	T2.2
BV5		81	T2.3
BV6		120	T2.4
BV7		121	T2.5
BV8		Long.MAX_VALUE	T2.6
BV9	goldCustomer	true	
BV10		false	
BV11	Return Value	FULLPRICE	
BV12		DISCOUNT	
BV13		ERROR	

Completing the TCI table

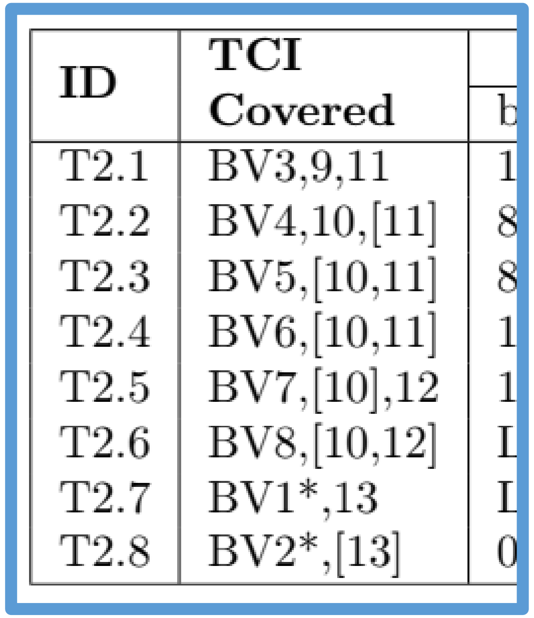


ID	TCI Covered	b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0

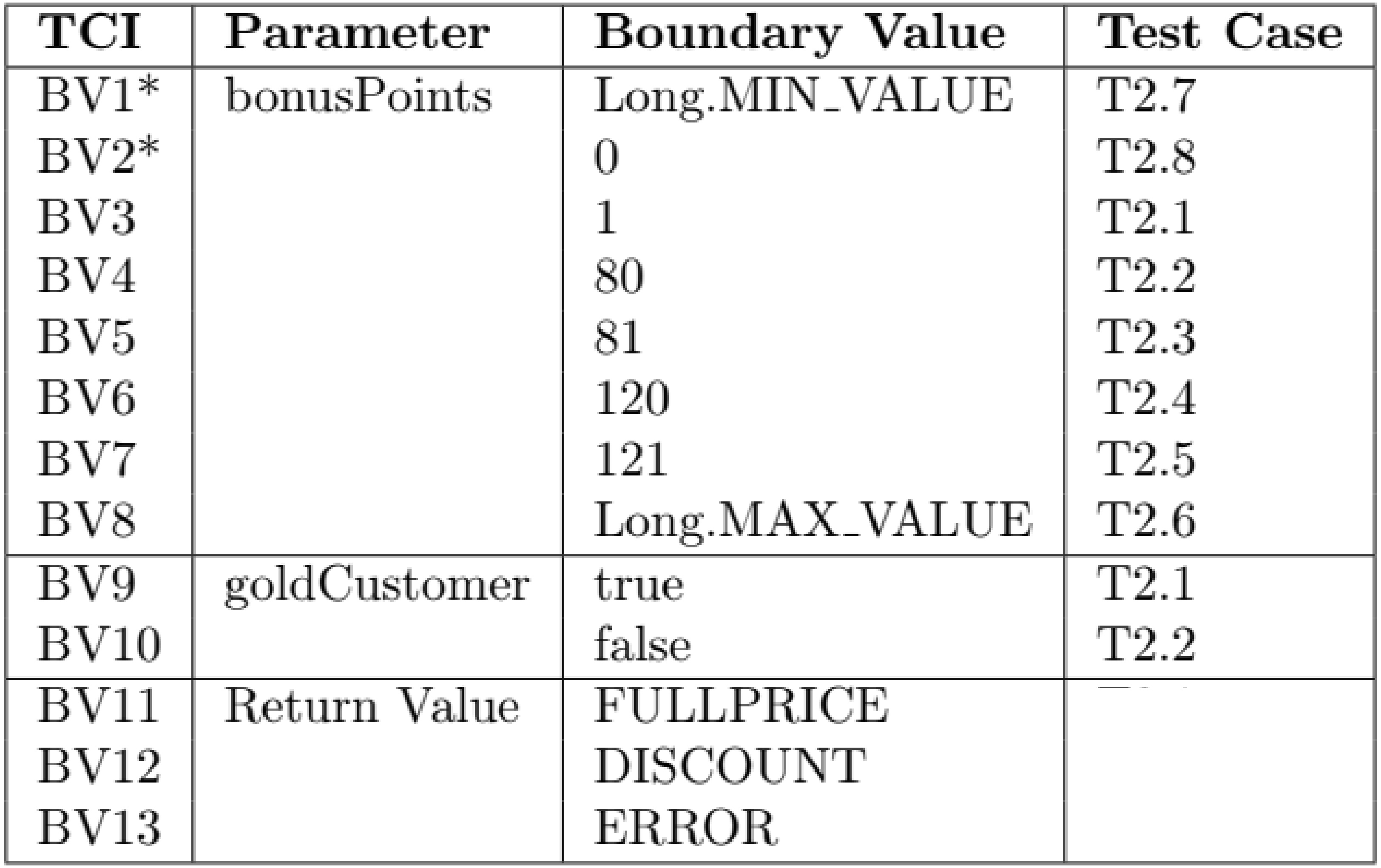


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	T2.8
BV3		1	T2.1
BV4		80	T2.2
BV5		81	T2.3
BV6		120	T2.4
BV7		121	T2.5
BV8		Long.MAX_VALUE	T2.6
BV9	goldCustomer	true	T2.1
BV10		false	
BV11	Return Value	FULLPRICE	
BV12		DISCOUNT	
BV13		ERROR	

Completing the TCI table

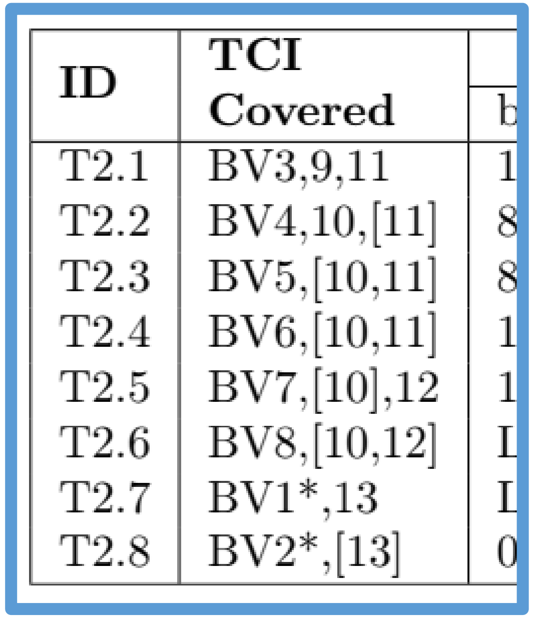


ID	TCI Covered	
		b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0

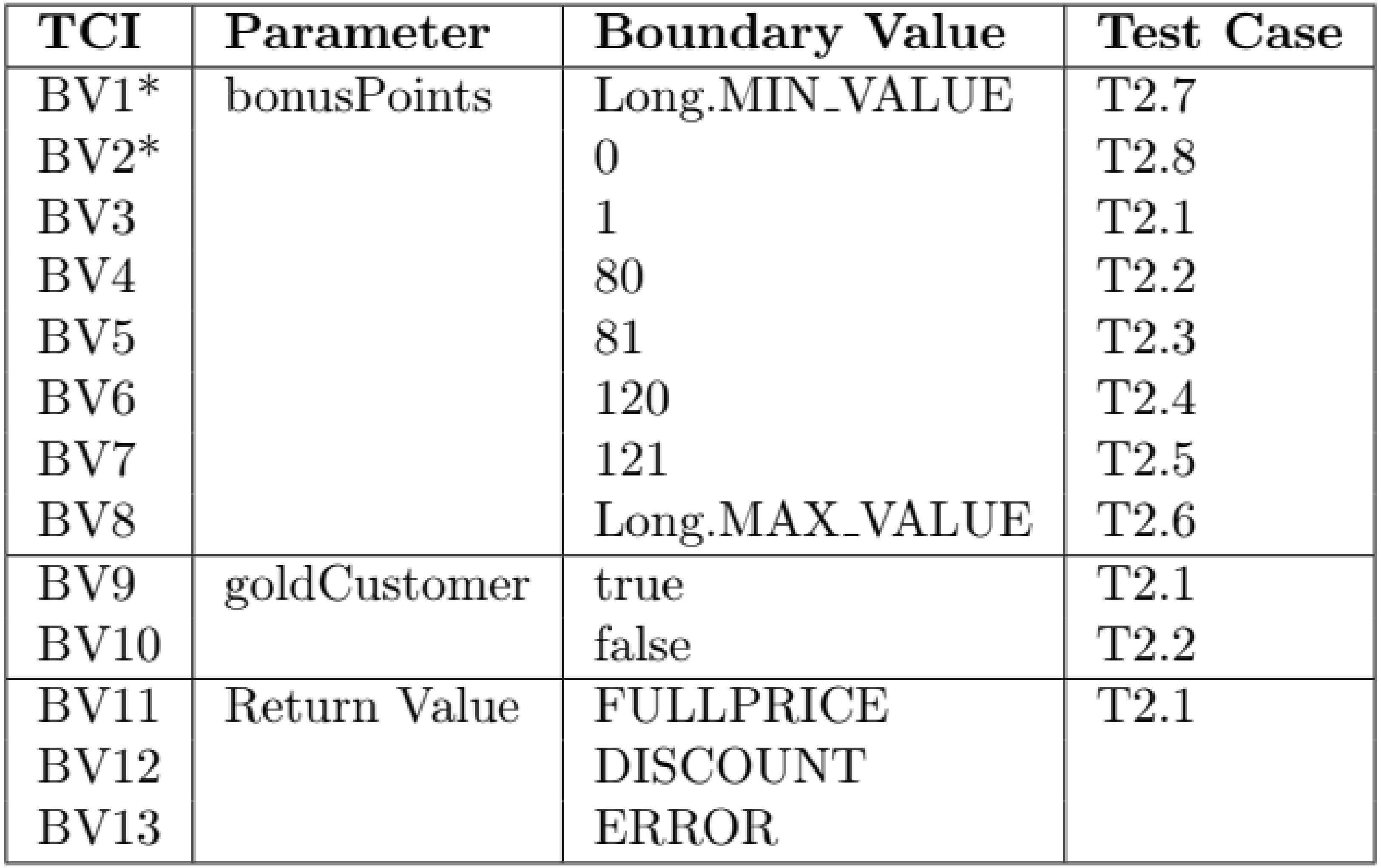


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	T2.8
BV3		1	T2.1
BV4		80	T2.2
BV5		81	T2.3
BV6		120	T2.4
BV7		121	T2.5
BV8		Long.MAX_VALUE	T2.6
BV9	goldCustomer	true	T2.1
BV10		false	T2.2
BV11	Return Value	FULLPRICE	
BV12		DISCOUNT	
BV13		ERROR	

Completing the TCI table

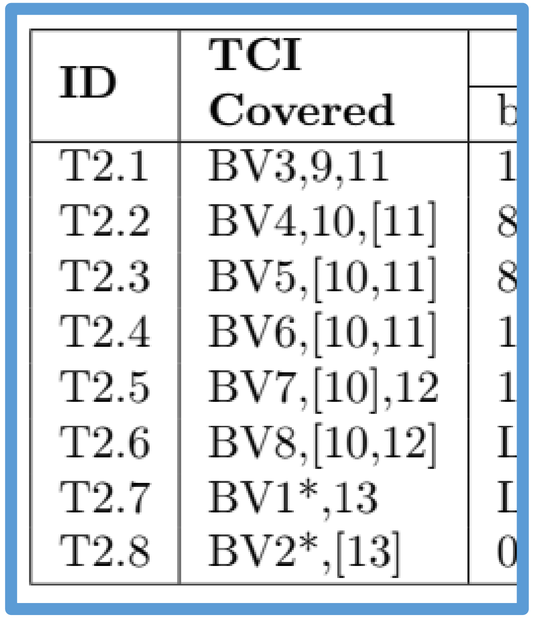


ID	TCI Covered	
		b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0

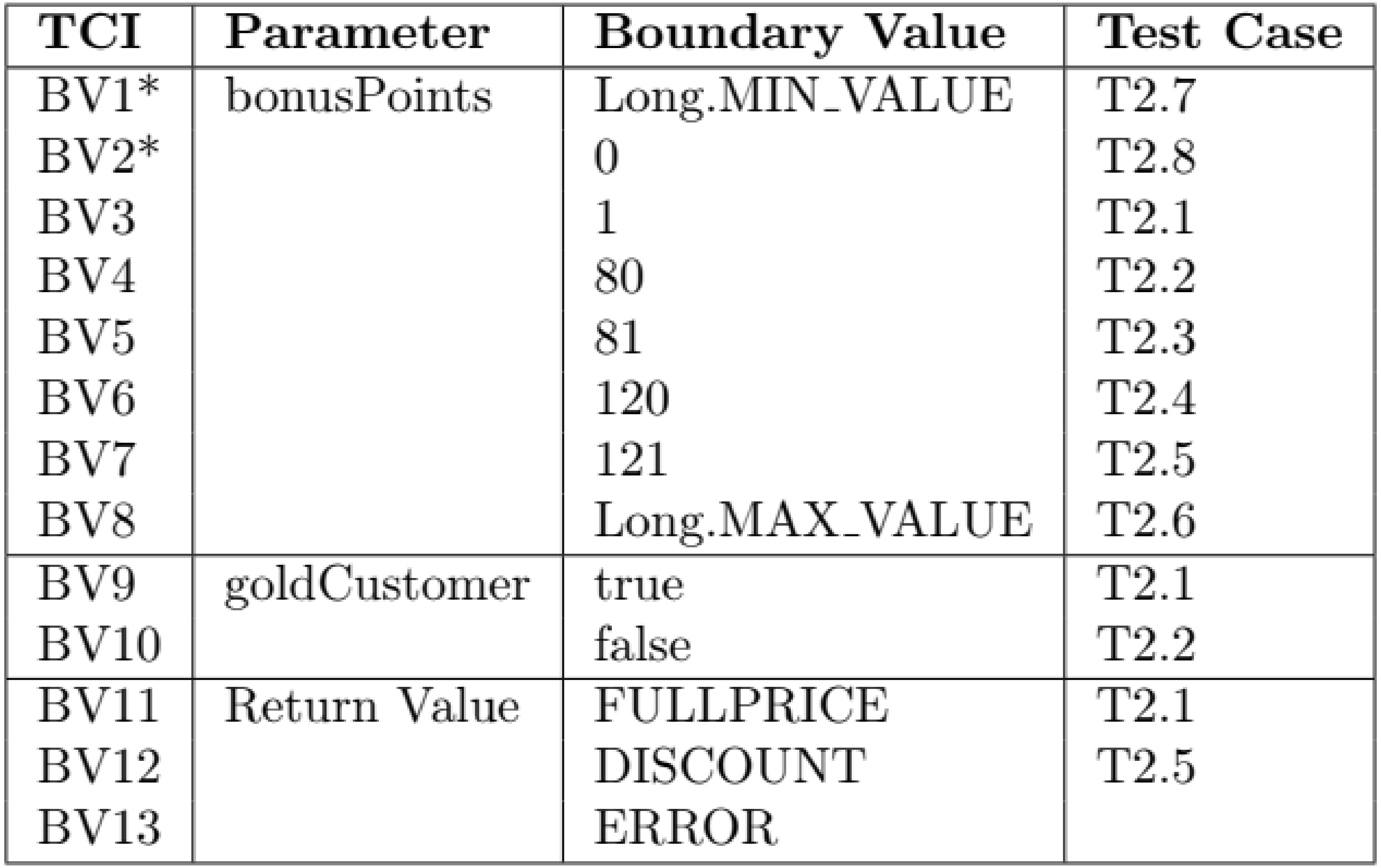


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	T2.8
BV3		1	T2.1
BV4		80	T2.2
BV5		81	T2.3
BV6		120	T2.4
BV7		121	T2.5
BV8		Long.MAX_VALUE	T2.6
BV9	goldCustomer	true	T2.1
BV10		false	T2.2
BV11	Return Value	FULLPRICE	T2.1
BV12		DISCOUNT	
BV13		ERROR	

Completing the TCI table

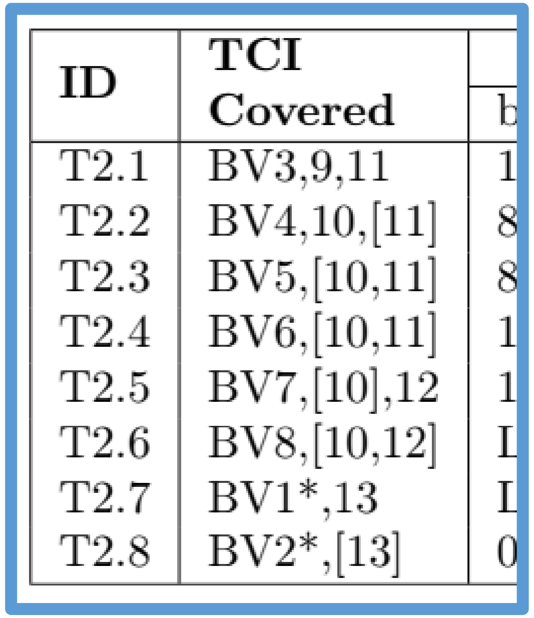


ID	TCI Covered	
		b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0

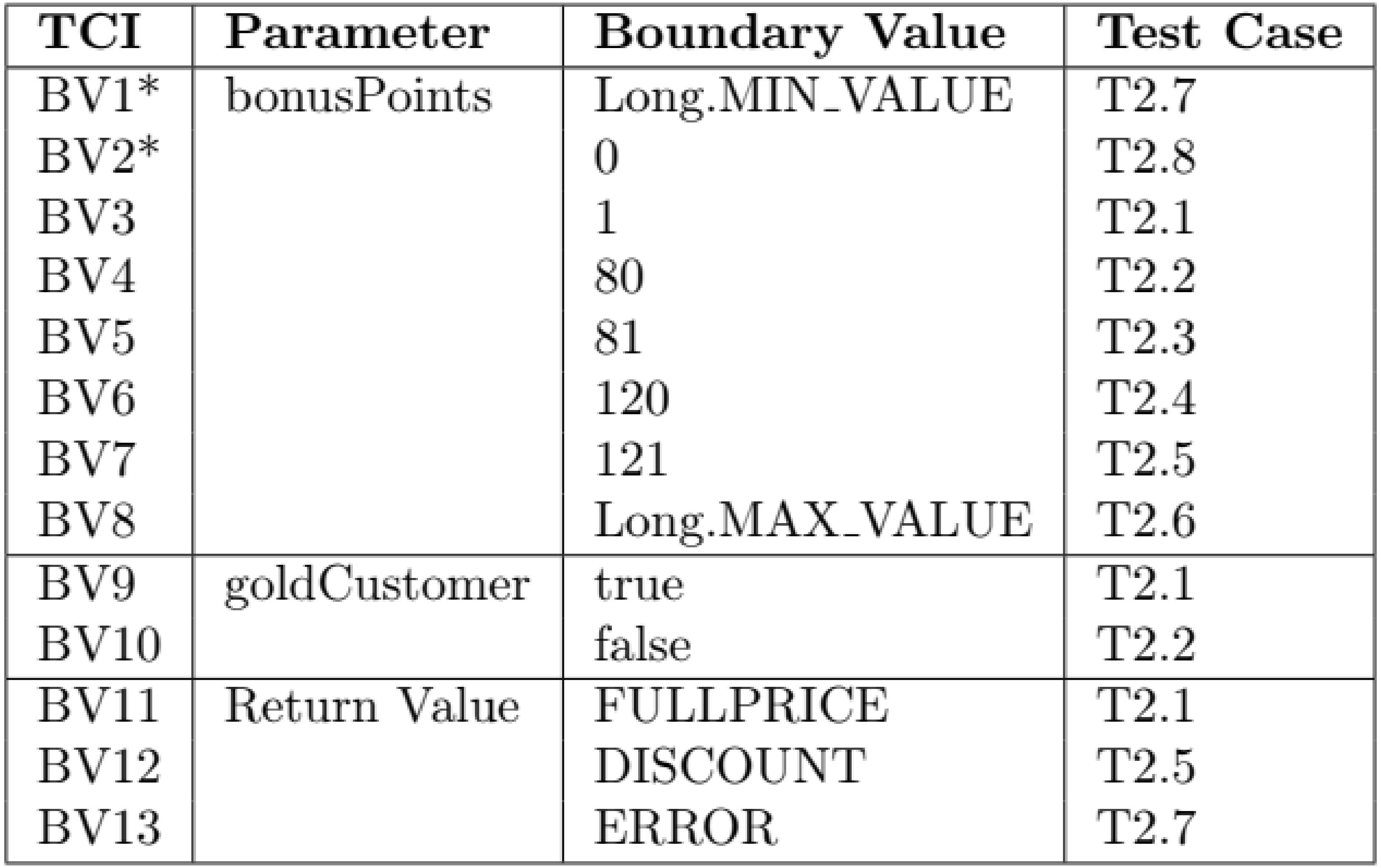


TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	T2.8
BV3		1	T2.1
BV4		80	T2.2
BV5		81	T2.3
BV6		120	T2.4
BV7		121	T2.5
BV8		Long.MAX_VALUE	T2.6
BV9	goldCustomer	true	T2.1
BV10		false	T2.2
BV11	Return Value	FULLPRICE	T2.1
BV12		DISCOUNT	T2.5
BV13		ERROR	

Completed Test Coverage Item Table



ID	TCI Covered	b
T2.1	BV3,9,11	1
T2.2	BV4,10,[11]	8
T2.3	BV5,[10,11]	8
T2.4	BV6,[10,11]	1
T2.5	BV7,[10],12	1
T2.6	BV8,[10,12]	L
T2.7	BV1*,13	L
T2.8	BV2*,[13]	0



TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	T2.8
BV3		1	T2.1
BV4		80	T2.2
BV5		81	T2.3
BV6		120	T2.4
BV7		121	T2.5
BV8		Long.MAX_VALUE	T2.6
BV9	goldCustomer	true	T2.1
BV10		false	T2.2
BV11	Return Value	FULLPRICE	T2.1
BV12		DISCOUNT	T2.5
BV13		ERROR	T2.7

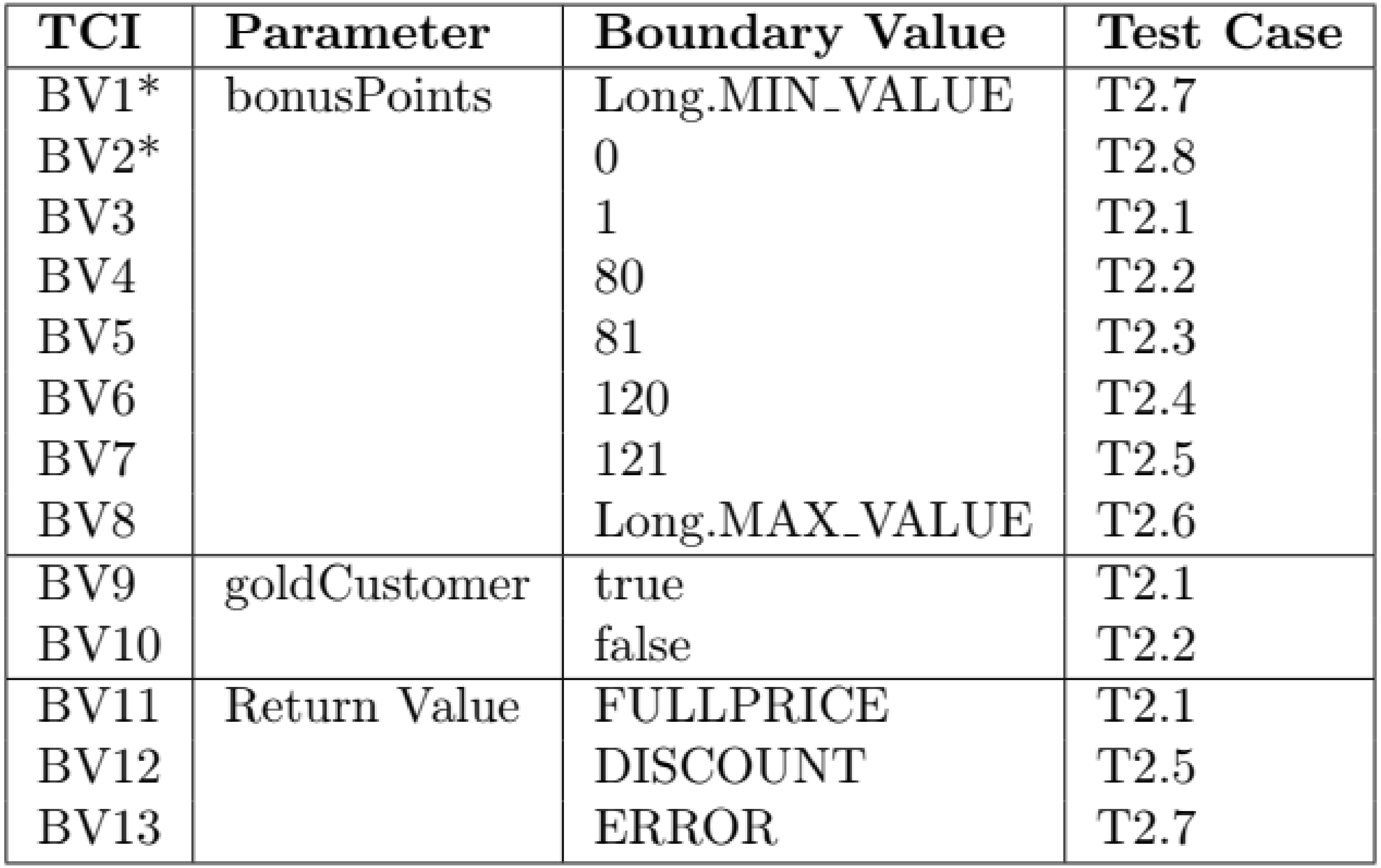
Reviewing Your Work

1. Every BVA test coverage item is covered by at least one test case
   - This confirms that the test cases are complete
2. Every new BVA test case covers at least one additional test coverage item:
   - this confirms that there are no unnecessary test cases
   - Ideally, each test case should cover as many new test coverage items as possible (up to three in this example: two input TCIs, and one output TCI)

- There should be no duplicate tests while taking the equivalence partition test cases into consideration

Reviewing the Test Coverage Items

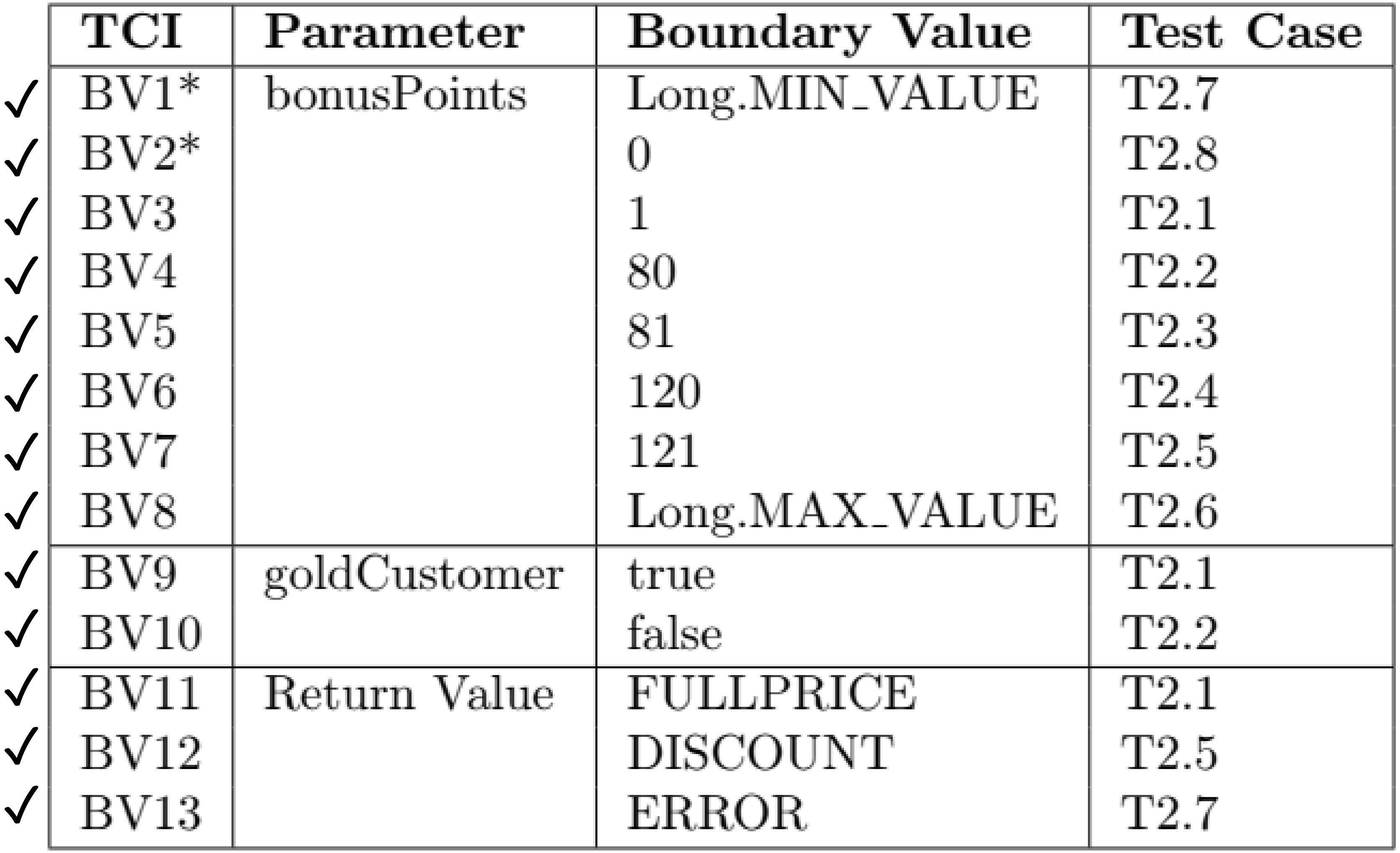
- Every TCI covered?



TCI	Parameter	Boundary Value	Test Case
BV1*	bonusPoints	Long.MIN_VALUE	T2.7
BV2*		0	T2.8
BV3		1	T2.1
BV4		80	T2.2
BV5		81	T2.3
BV6		120	T2.4
BV7		121	T2.5
BV8		Long.MAX_VALUE	T2.6
BV9	goldCustomer	true	T2.1
BV10		false	T2.2
BV11	Return Value	FULLPRICE	T2.1
BV12		DISCOUNT	T2.5
BV13		ERROR	T2.7

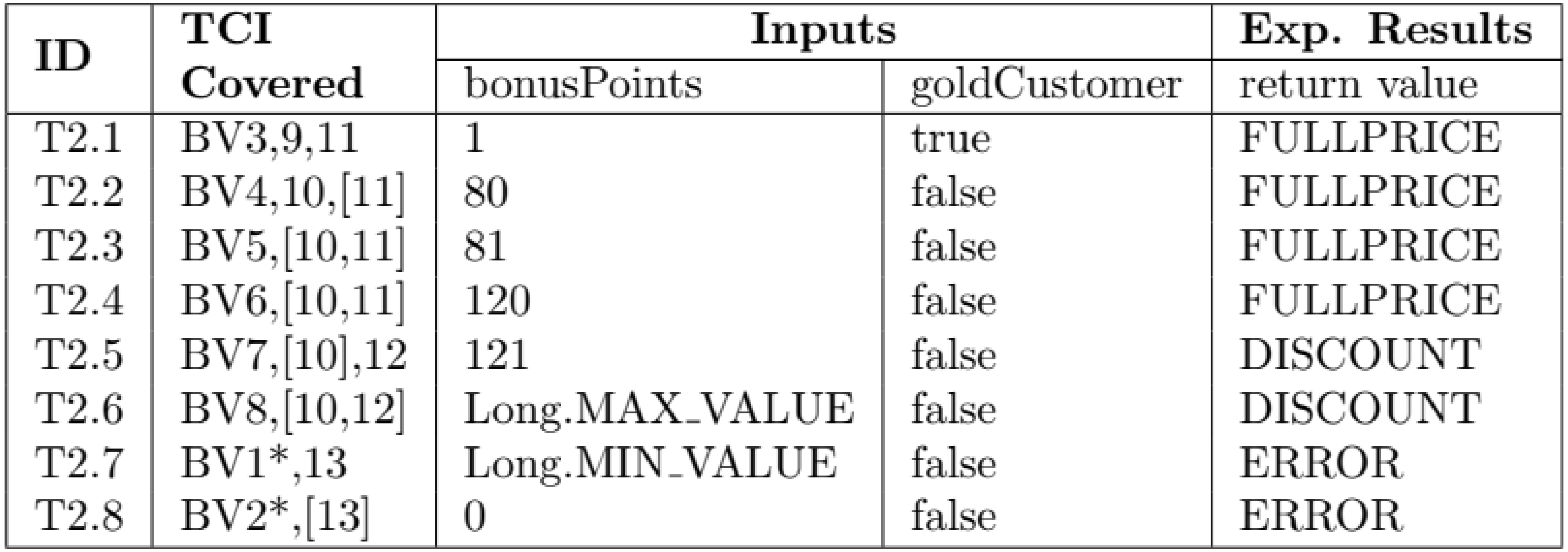
Reviewing the Test Coverage Items

- Every TCI is covered by at least one test case



	TCI	Parameter	Boundary Value	Test Case
✓	BV1*	bonusPoints	Long.MIN_VALUE	T2.7
✓	BV2*		0	T2.8
✓	BV3		1	T2.1
✓	BV4		80	T2.2
✓	BV5		81	T2.3
✓	BV6		120	T2.4
✓	BV7		121	T2.5
✓	BV8		Long.MAX_VALUE	T2.6
✓	BV9	goldCustomer	true	T2.1
✓	BV10		false	T2.2
✓	BV11	Return Value	FULLPRICE	T2.1
✓	BV12		DISCOUNT	T2.5
✓	BV13		ERROR	T2.7

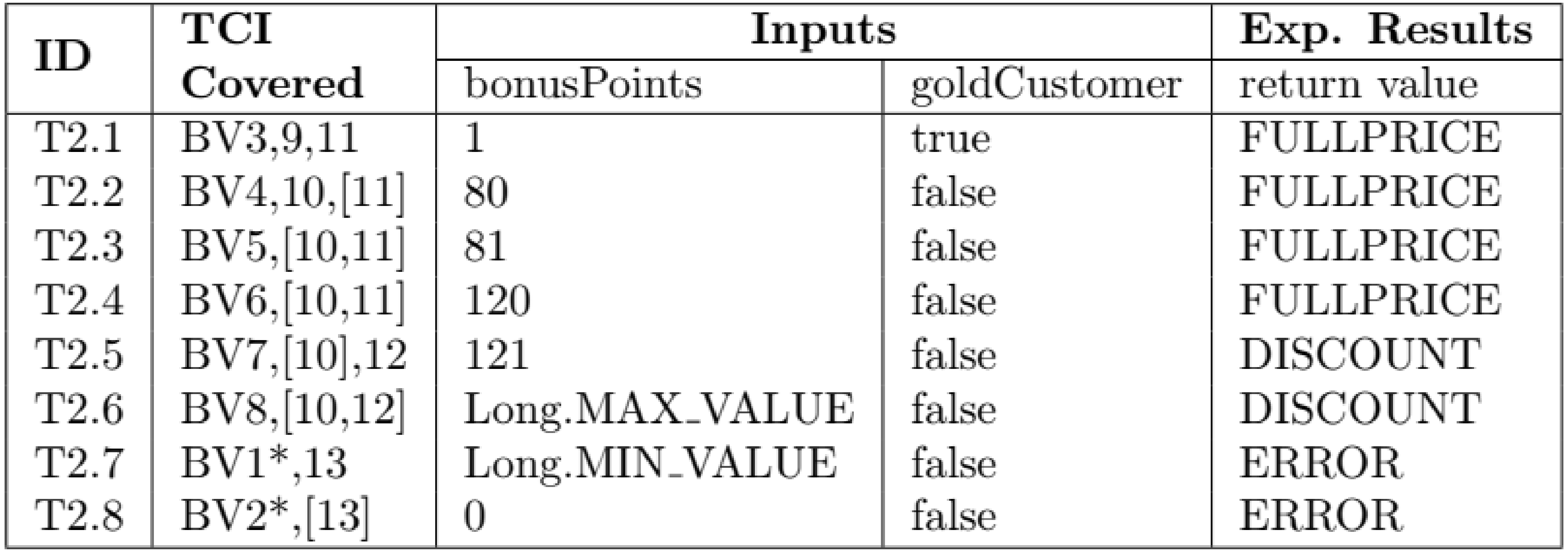
Reviewing the Test Cases



ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

- T2.1 covers BV3, BV9, and BV11 for the first time

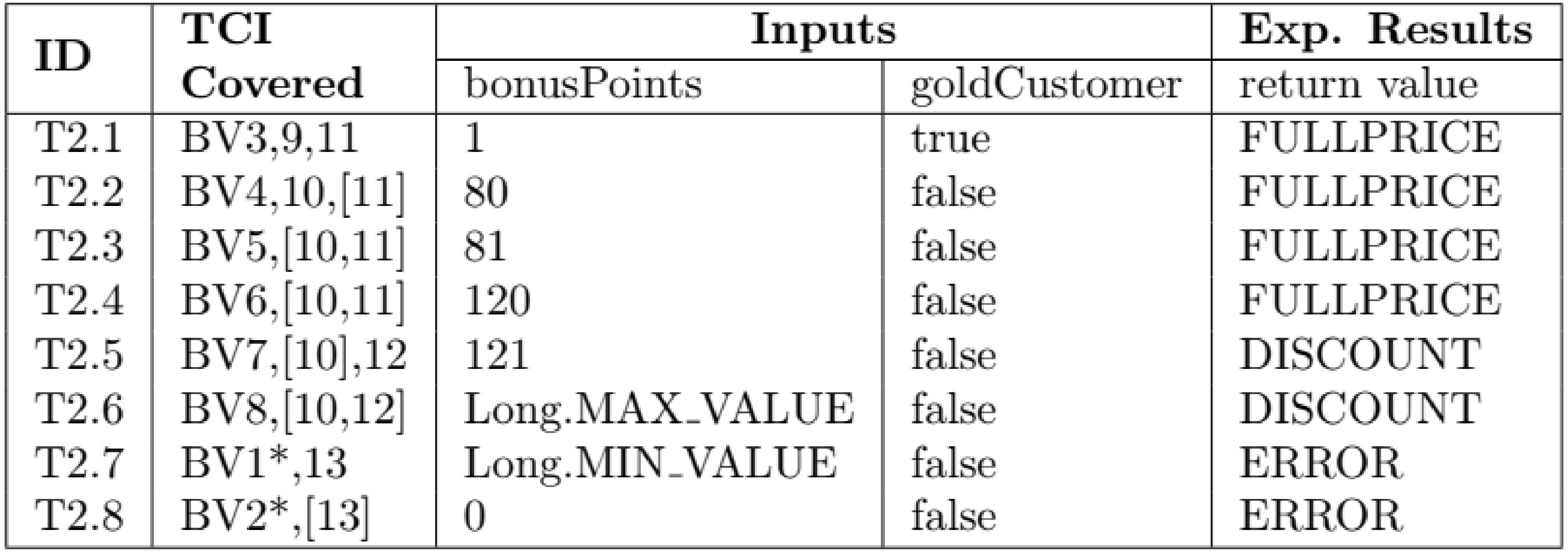
Reviewing the Test Cases



ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

- T2.1 covers BV3, BV9, and BV11 for the first time
- T2.2 covers BV4 and BV10 for the first time (covering BV11 again is unavoidable)

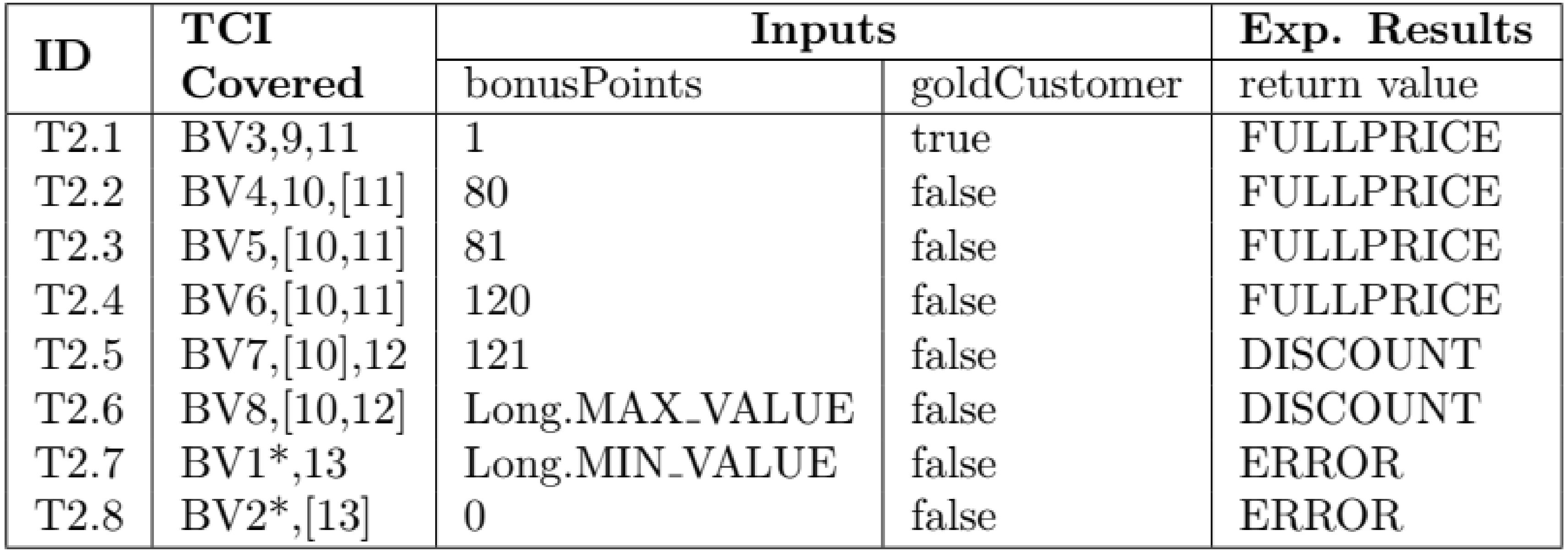
Reviewing the Test Cases



ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

- T2.1 covers BV3, BV9, and BV11 for the first time
- T2.2 covers BV4 and BV10 for the first time (covering BV11 again is unavoidable)
- T2.3 covers BV5 for the first time (covering BV10 and BV11 again is unavoidable)

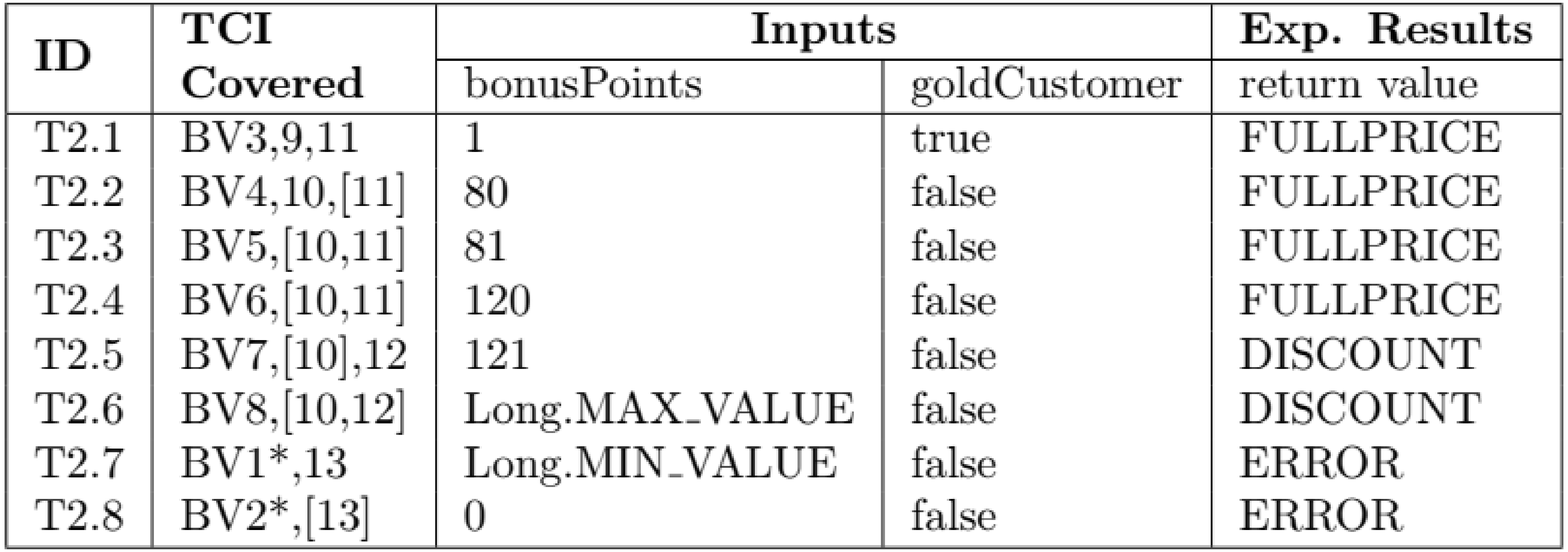
Reviewing the Test Cases



ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

- T2.1 covers BV3, BV9, and BV11 for the first time
- T2.2 covers BV4 and BV10 for the first time (covering BV11 again is unavoidable)
- T2.3 covers BV5 for the first time (covering BV10 and BV11 again is unavoidable)
- T2.4 covers BV6 for the first time (covering BV10 and BV11 again is unavoidable)

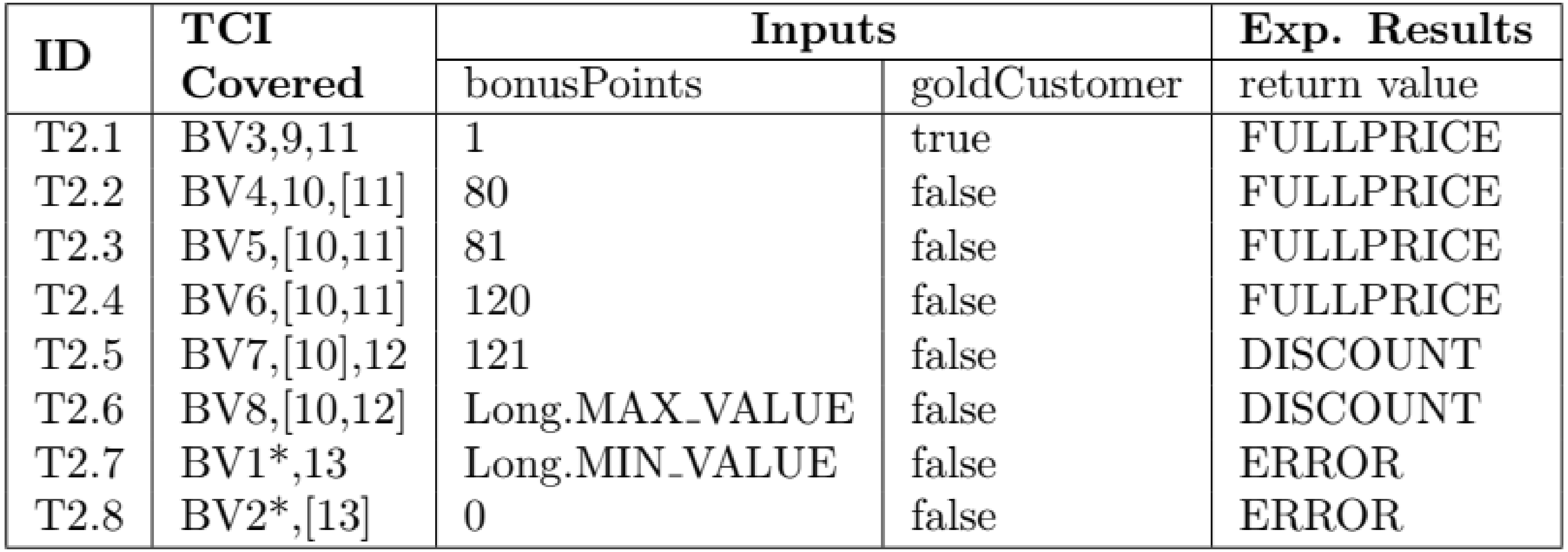
Reviewing the Test Cases



ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

- T2.1 covers BV3, BV9, and BV11 for the first time
- T2.2 covers BV4 and BV10 for the first time (covering BV11 again is unavoidable)
- T2.3 covers BV5 for the first time (covering BV10 and BV11 again is unavoidable)
- T2.4 covers BV6 for the first time (covering BV10 and BV11 again is unavoidable)
- T2.5 covers BV7 and BV12 (covering BV10 again is unavoidable)

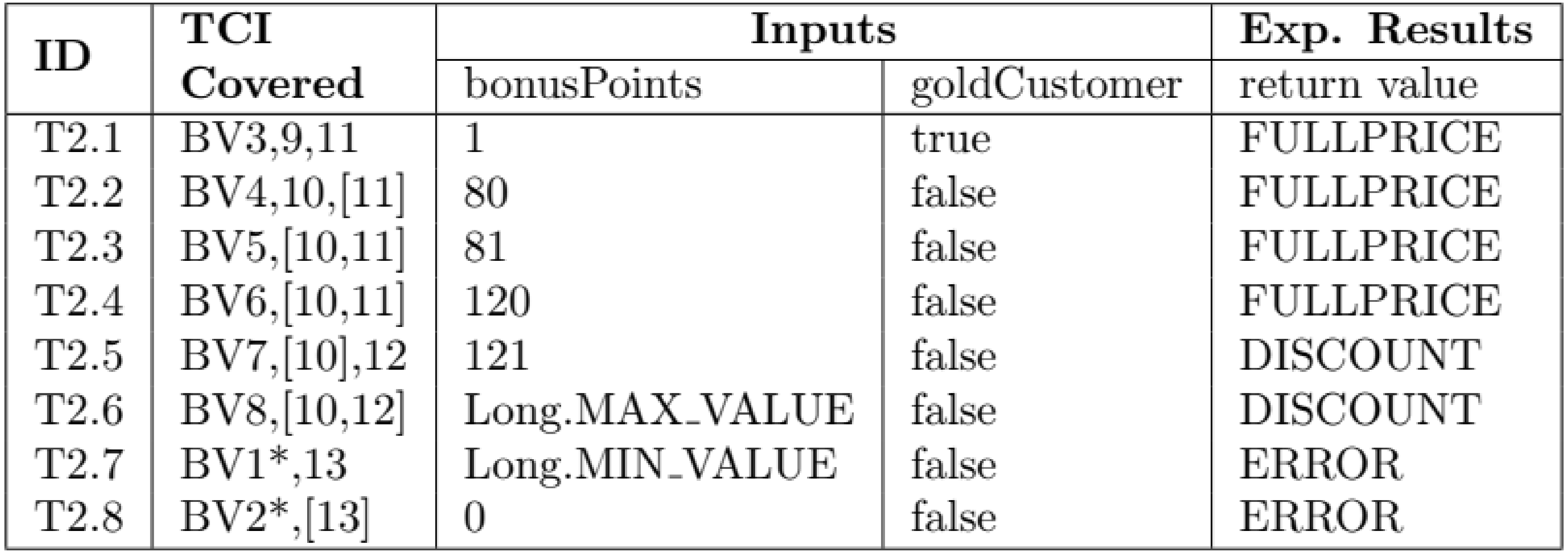
Reviewing the Test Cases



ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

- T2.1 covers BV3, BV9, and BV11 for the first time
- T2.2 covers BV4 and BV10 for the first time (covering BV11 again is unavoidable)
- T2.3 covers BV5 for the first time (covering BV10 and BV11 again is unavoidable)
- T2.4 covers BV6 for the first time (covering BV10 and BV11 again is unavoidable)
- T2.5 covers BV7 and BV12 (covering BV10 again is unavoidable)
- T2.6 covers BV8 (covering BV10 and BV12 again is unavoidable)

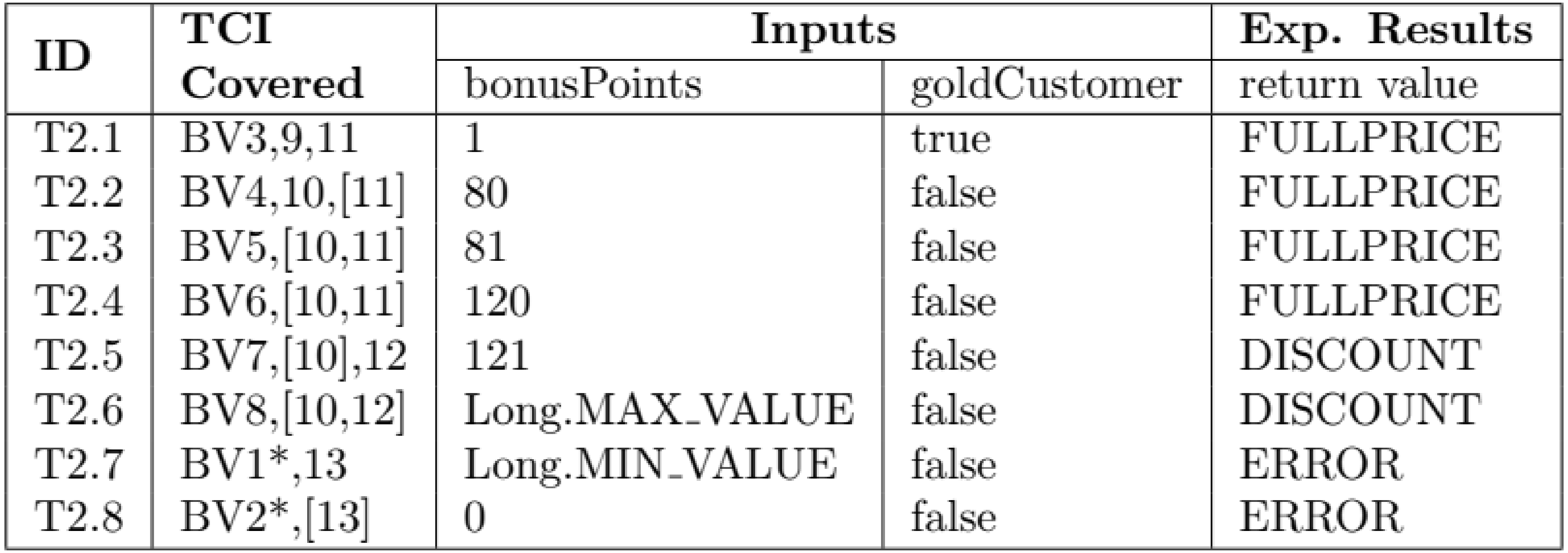
Reviewing the error Test Cases



ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

- T2.7 is an error test case, it covers the single input error test coverage item BV1*
  - It also covers the output test coverage item BV13
  - Although the selected input value of goldCustomer is false, it does not cover BV10 (error hiding)

Reviewing the error Test Cases



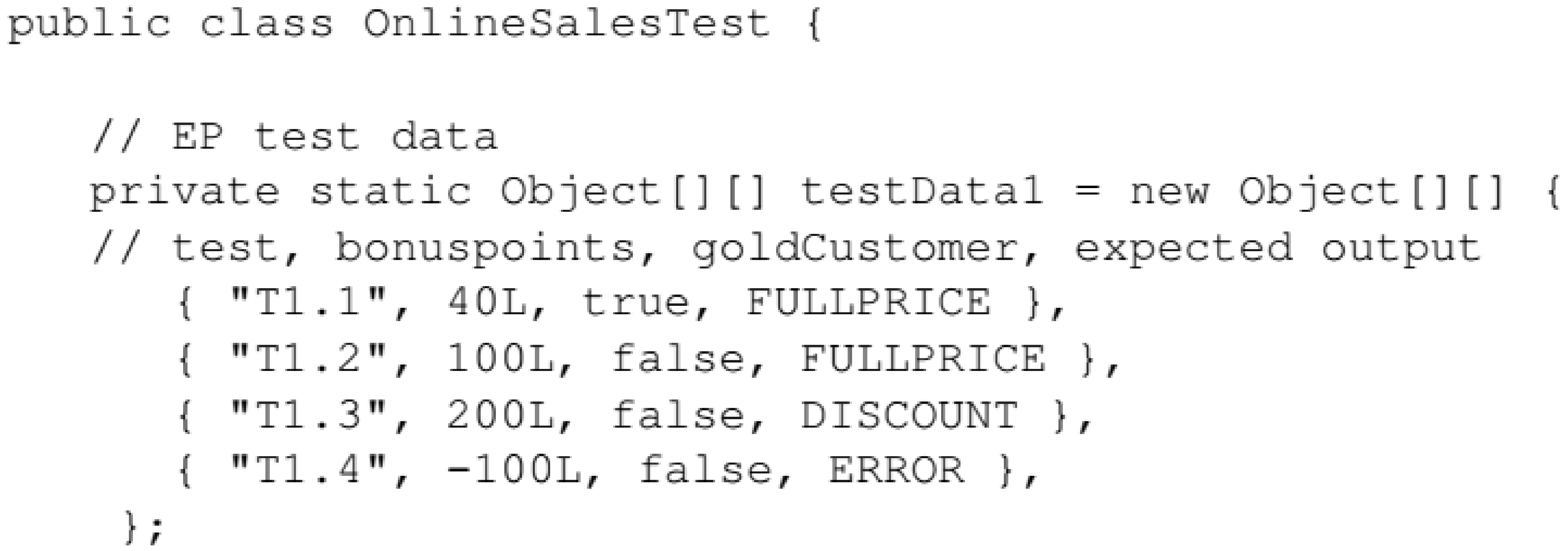
ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

- T2.7 is an error test case, it covers the single input error test coverage item BV1*
  - It also covers the output test coverage item BV13
  - Although the selected input value of goldCustomer is false, it does not cover BV10
- T2.8 is also an error test case, it covers BV2*
  - It also unavoidably covers BV13 again
  - As in the previous test case, it does not cover BV10

5. Implementation

- You can write a separate test class for the boundary value analysis tests
- **Usual practice to keep extending the existing test class**
- The full test implementation, including the previously developed equivalence partition tests, is shown next
- For brevity, the include statements are omitted

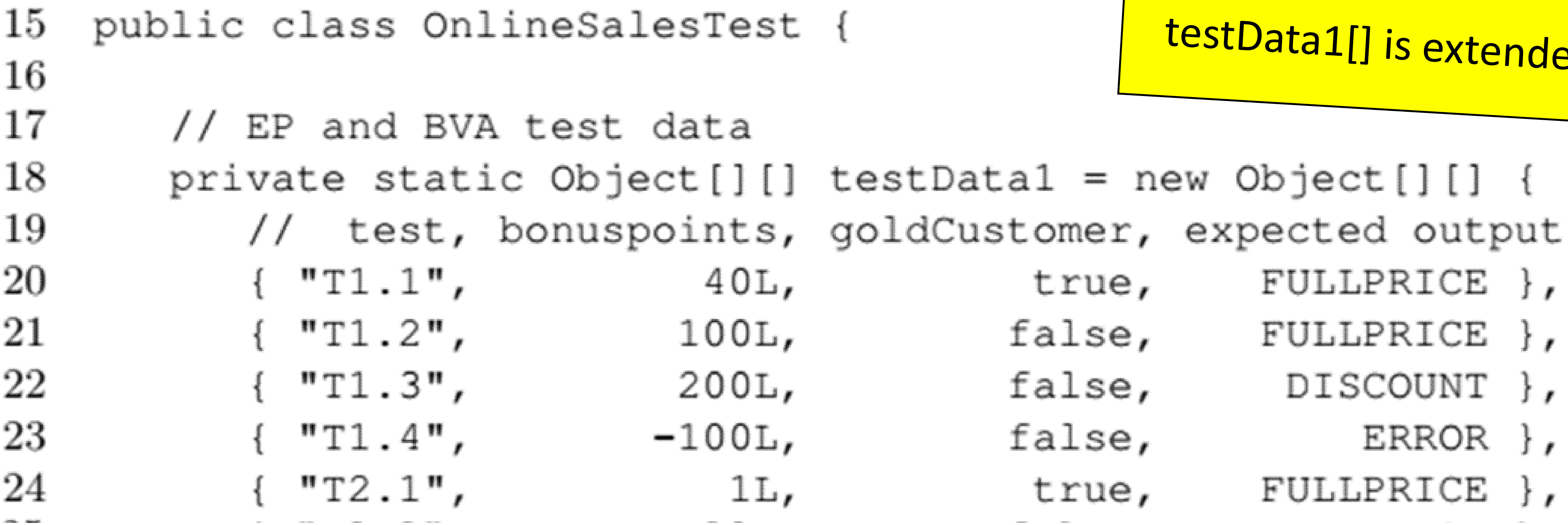
OnlineSalesTest with EP Coverage



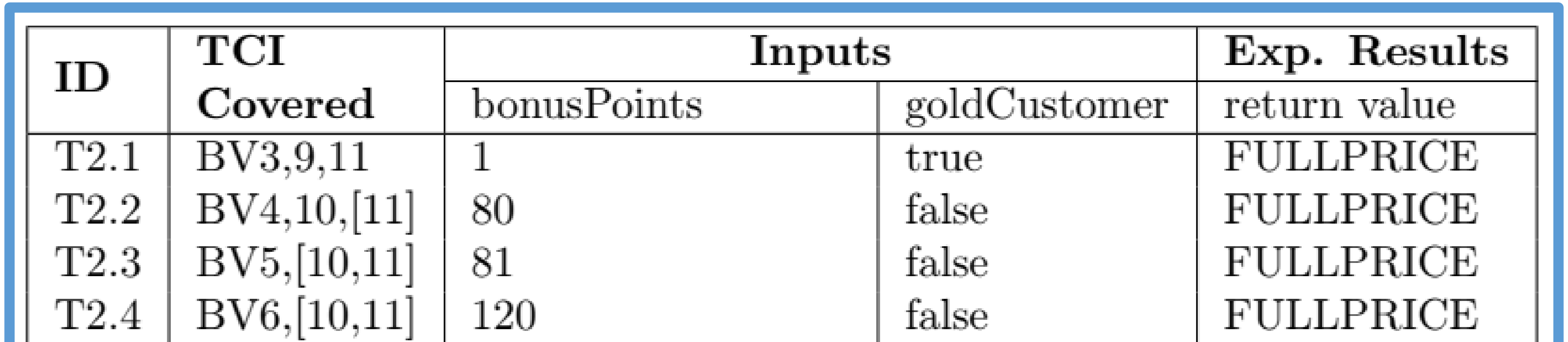
```
public class OnlineSalesTest {

   // EP test data
   private static Object[][] testData1 = new Object[][] {
   // test, bonuspoints, goldCustomer, expected output
      { "T1.1", 40L, true, FULLPRICE },
      { "T1.2", 100L, false, FULLPRICE },
      { "T1.3", 200L, false, DISCOUNT },
      { "T1.4", -100L, false, ERROR },
    };
```

OnlineSalesTest with BVA Coverage

testData1[] is extended



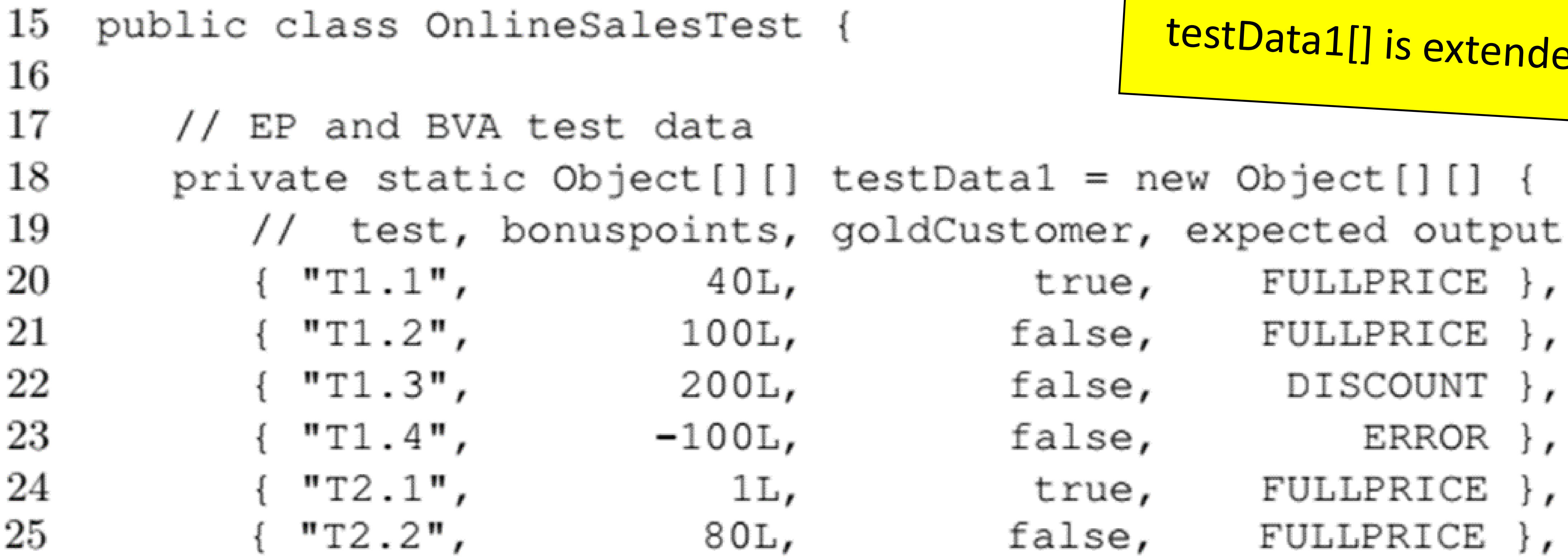
```
public class OnlineSalesTest {

    // EP and BVA test data
    private static Object[][] testData1 = new Object[][] {
        //  test, bonuspoints, goldCustomer, expected output
        { "T1.1",          40L,          true,    FULLPRICE },
        { "T1.2",         100L,         false,    FULLPRICE },
        { "T1.3",         200L,         false,     DISCOUNT },
        { "T1.4",        -100L,         false,        ERROR },
        { "T2.1",           1L,          true,    FULLPRICE },
```



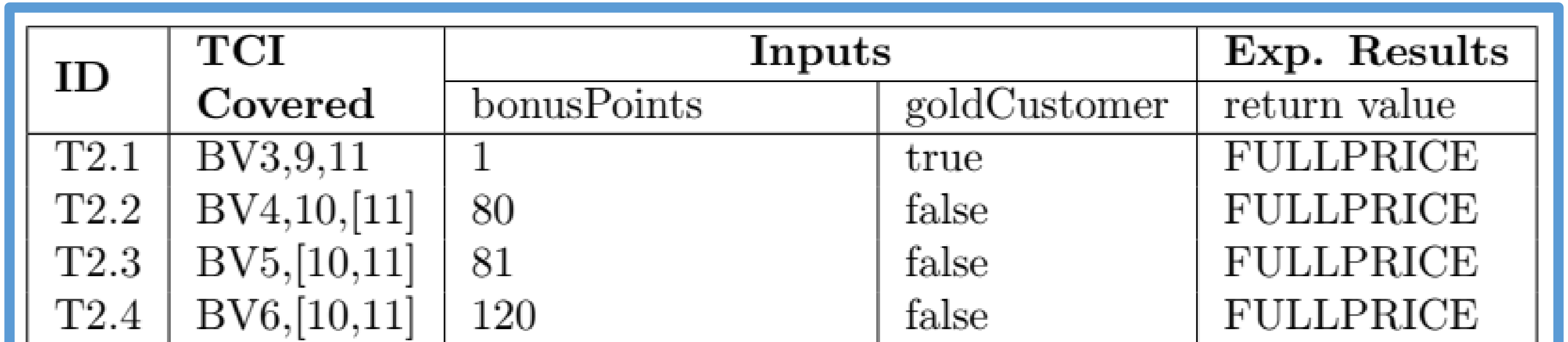
ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE

OnlineSalesTest with BVA Coverage

testData1[] is extended



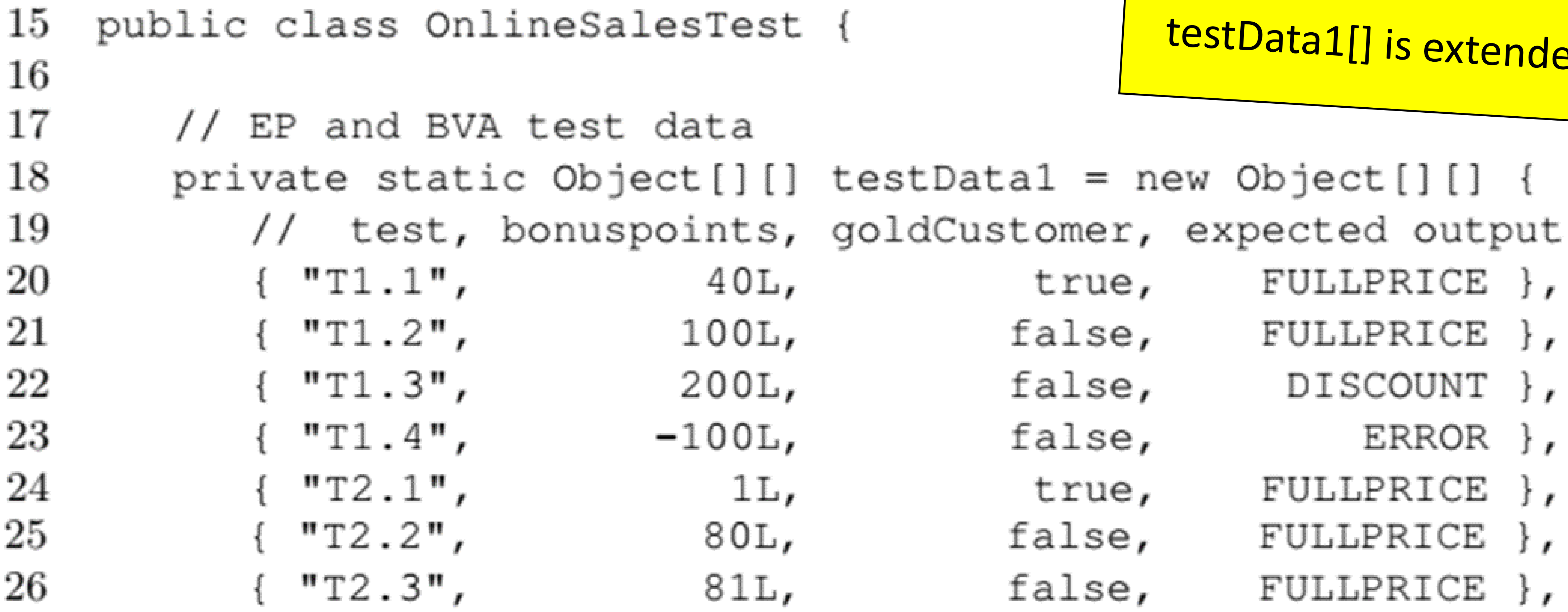
```
public class OnlineSalesTest {

    // EP and BVA test data
    private static Object[][] testData1 = new Object[][] {
       //  test, bonuspoints, goldCustomer, expected output
       { "T1.1",         40L,         true,    FULLPRICE },
       { "T1.2",        100L,        false,    FULLPRICE },
       { "T1.3",        200L,        false,     DISCOUNT },
       { "T1.4",       -100L,        false,        ERROR },
       { "T2.1",          1L,         true,    FULLPRICE },
       { "T2.2",         80L,        false,    FULLPRICE },
```



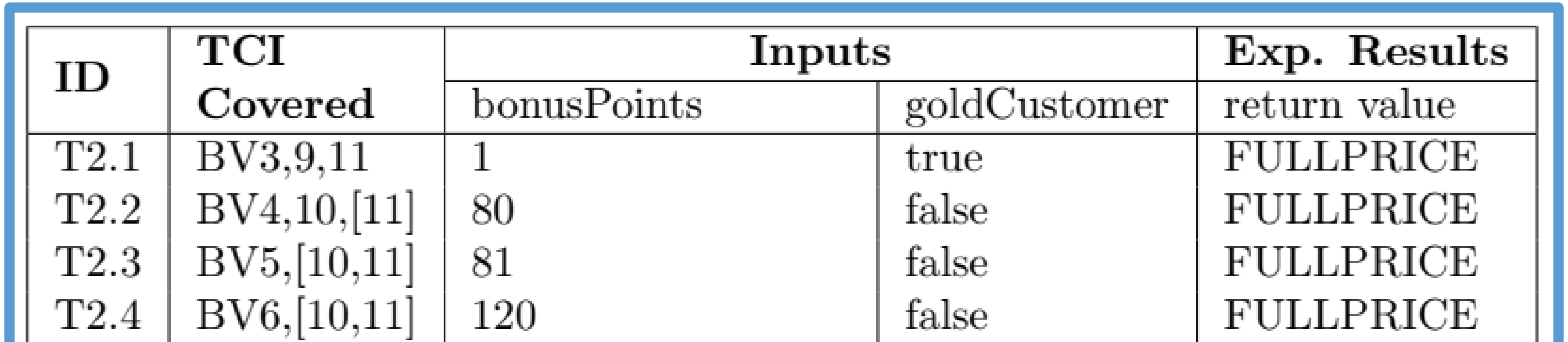
ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE

OnlineSalesTest with BVA Coverage

testData1[] is extended



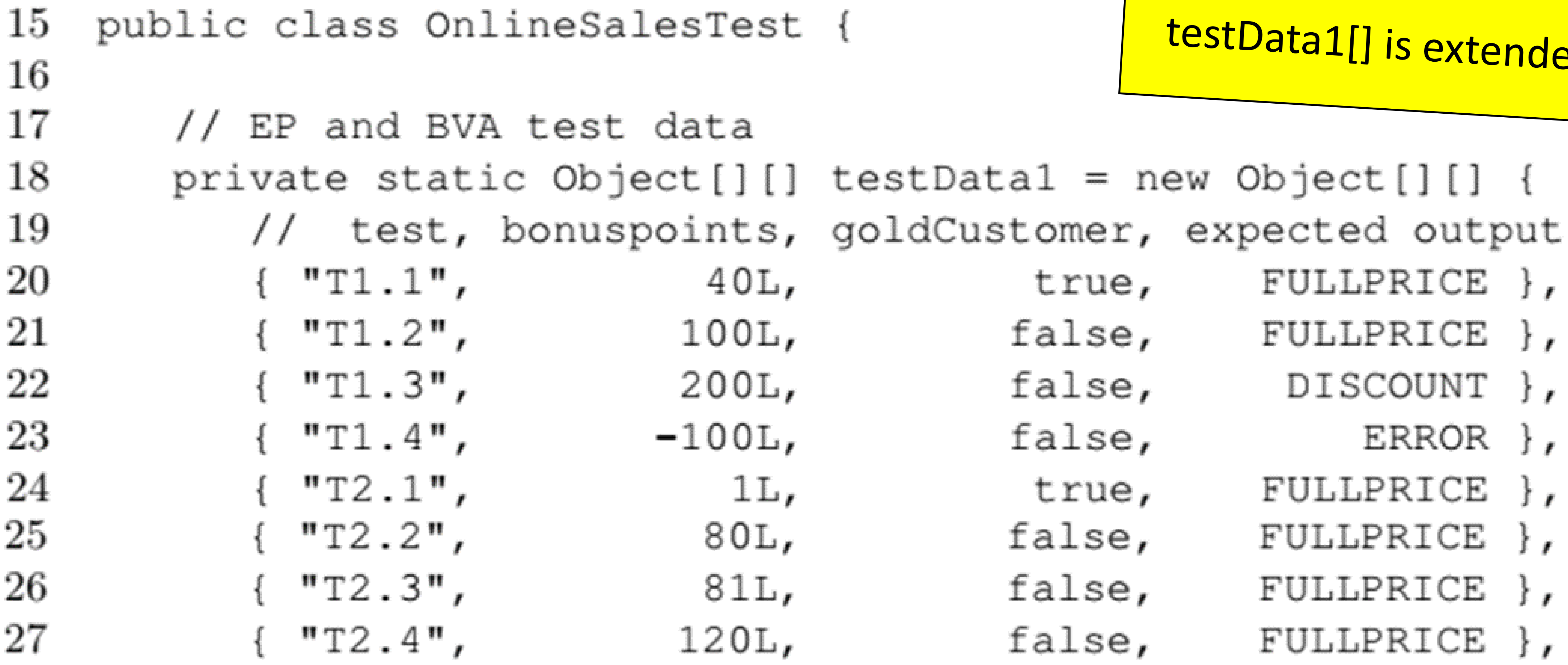
```
public class OnlineSalesTest {

   // EP and BVA test data
   private static Object[][] testData1 = new Object[][] {
      //  test, bonuspoints, goldCustomer, expected output
      { "T1.1",         40L,         true,    FULLPRICE },
      { "T1.2",        100L,        false,    FULLPRICE },
      { "T1.3",        200L,        false,     DISCOUNT },
      { "T1.4",       -100L,        false,        ERROR },
      { "T2.1",          1L,         true,    FULLPRICE },
      { "T2.2",         80L,        false,    FULLPRICE },
      { "T2.3",         81L,        false,    FULLPRICE },
```



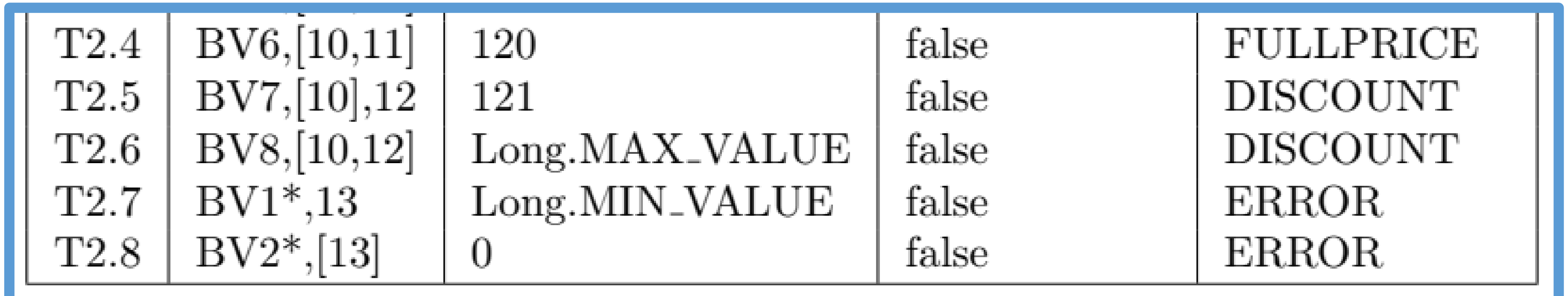
ID	TCI Covered	Inputs		Exp. Results
		bonusPoints	goldCustomer	return value
T2.1	BV3,9,11	1	true	FULLPRICE
T2.2	BV4,10,[11]	80	false	FULLPRICE
T2.3	BV5,[10,11]	81	false	FULLPRICE
T2.4	BV6,[10,11]	120	false	FULLPRICE

OnlineSalesTest with BVA Coverage

testData1[] is extended



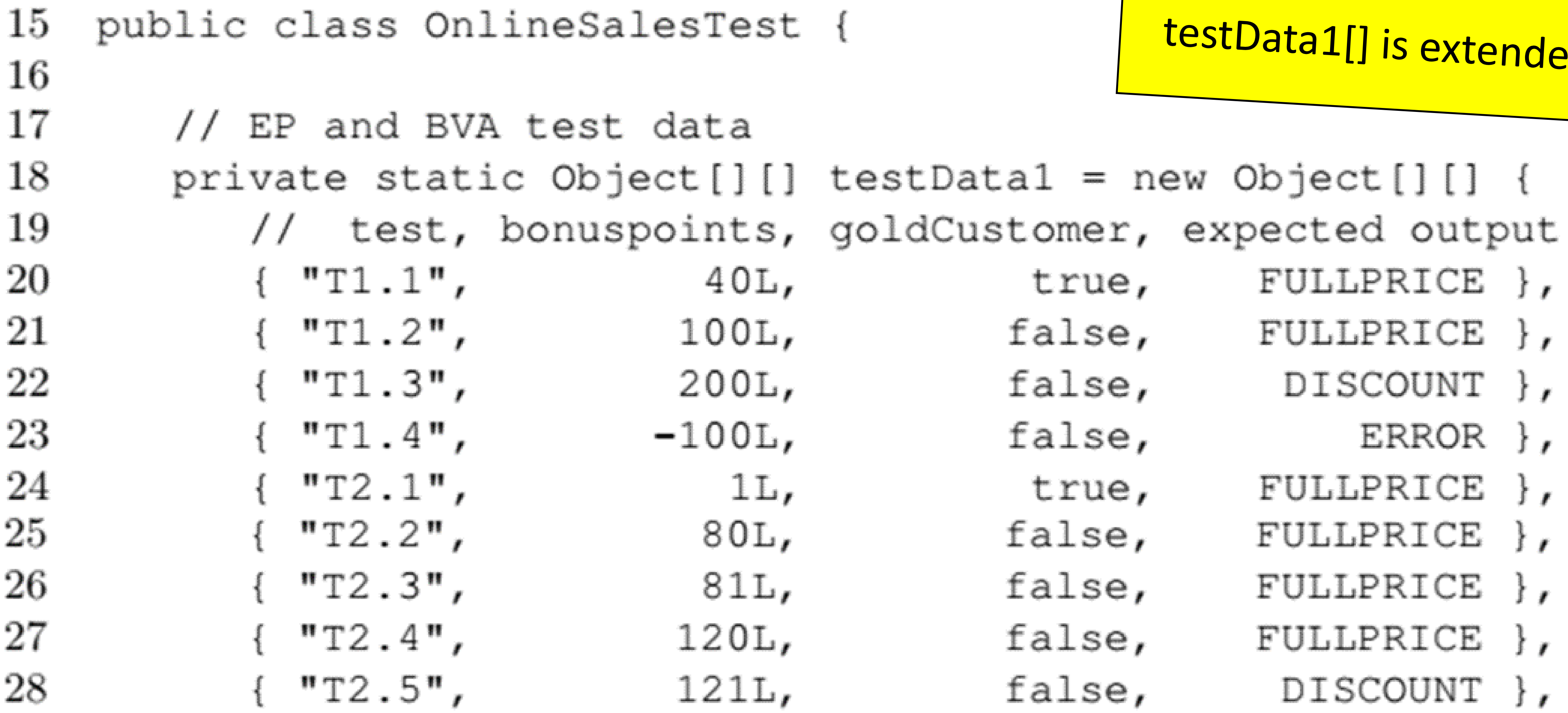
```
public class OnlineSalesTest {

   // EP and BVA test data
   private static Object[][] testData1 = new Object[][] {
      //  test, bonuspoints, goldCustomer, expected output
      { "T1.1",        40L,          true,   FULLPRICE },
      { "T1.2",       100L,         false,   FULLPRICE },
      { "T1.3",       200L,         false,    DISCOUNT },
      { "T1.4",      -100L,         false,       ERROR },
      { "T2.1",         1L,          true,   FULLPRICE },
      { "T2.2",        80L,         false,   FULLPRICE },
      { "T2.3",        81L,         false,   FULLPRICE },
      { "T2.4",       120L,         false,   FULLPRICE },
```



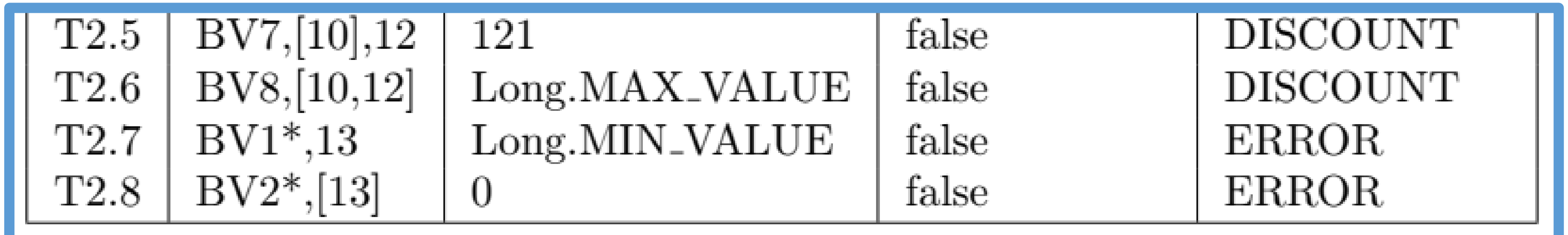
T2.4	BV6,[10,11]	120	false	FULLPRICE
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

OnlineSalesTest with BVA Coverage

testData1[] is extended



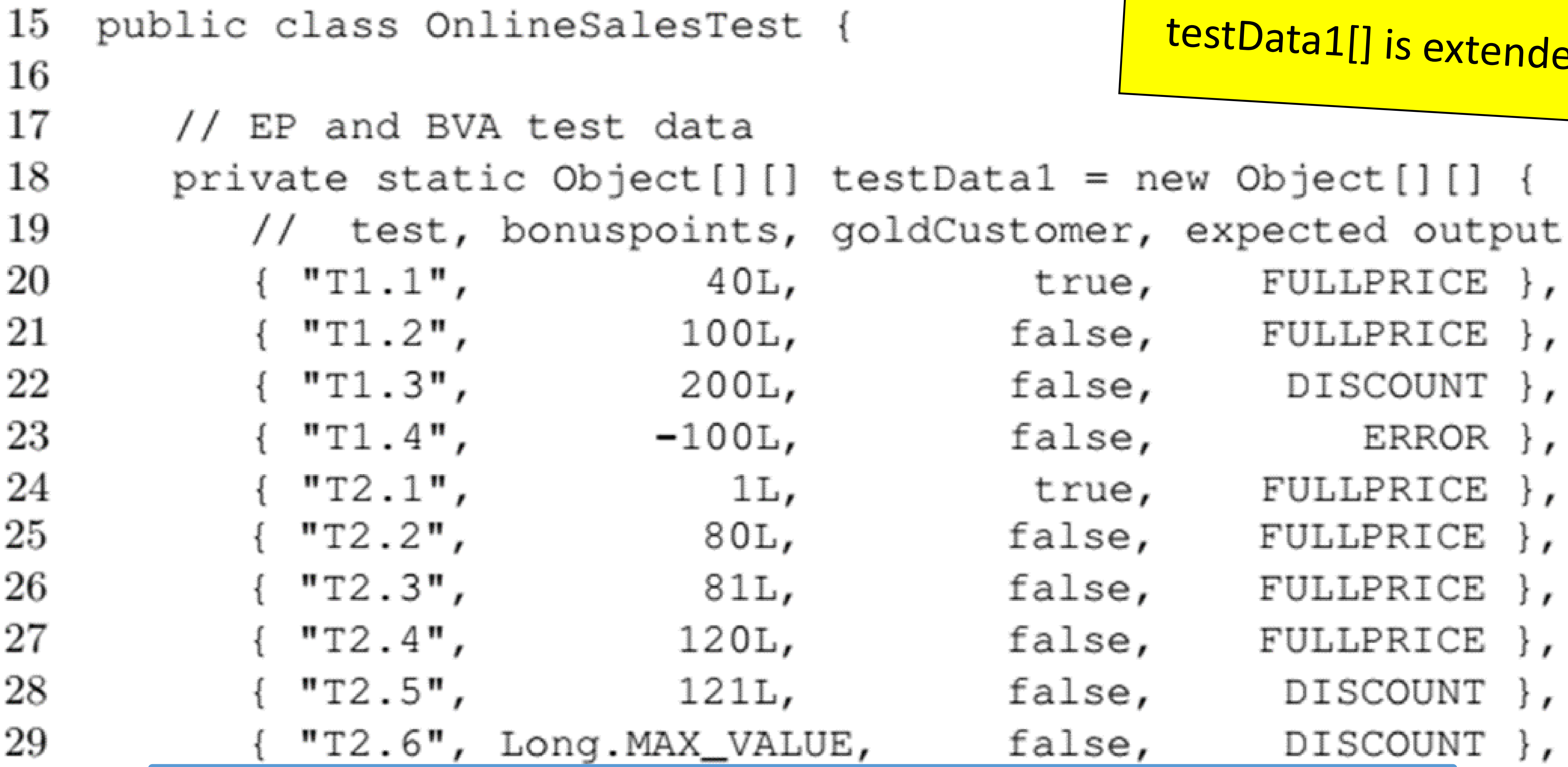
```
public class OnlineSalesTest {

   // EP and BVA test data
   private static Object[][] testData1 = new Object[][] {
      //  test, bonuspoints, goldCustomer, expected output
      { "T1.1",         40L,         true,    FULLPRICE },
      { "T1.2",        100L,        false,    FULLPRICE },
      { "T1.3",        200L,        false,     DISCOUNT },
      { "T1.4",       -100L,        false,        ERROR },
      { "T2.1",          1L,         true,    FULLPRICE },
      { "T2.2",         80L,        false,    FULLPRICE },
      { "T2.3",         81L,        false,    FULLPRICE },
      { "T2.4",        120L,        false,    FULLPRICE },
      { "T2.5",        121L,        false,     DISCOUNT },
```



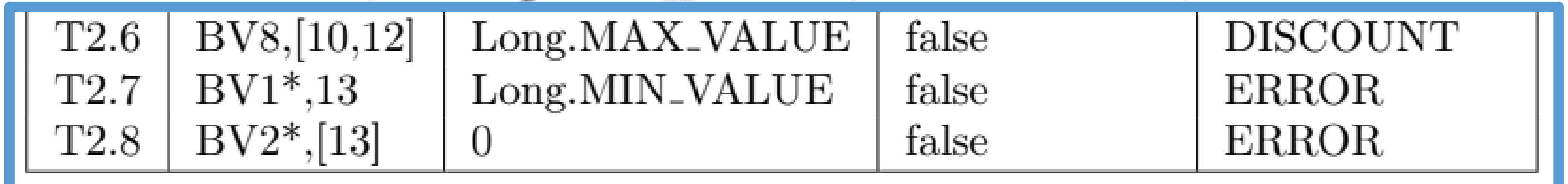
T2.5	BV7,[10],12	121	false	DISCOUNT
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

OnlineSalesTest with BVA Coverage

testData1[] is extended



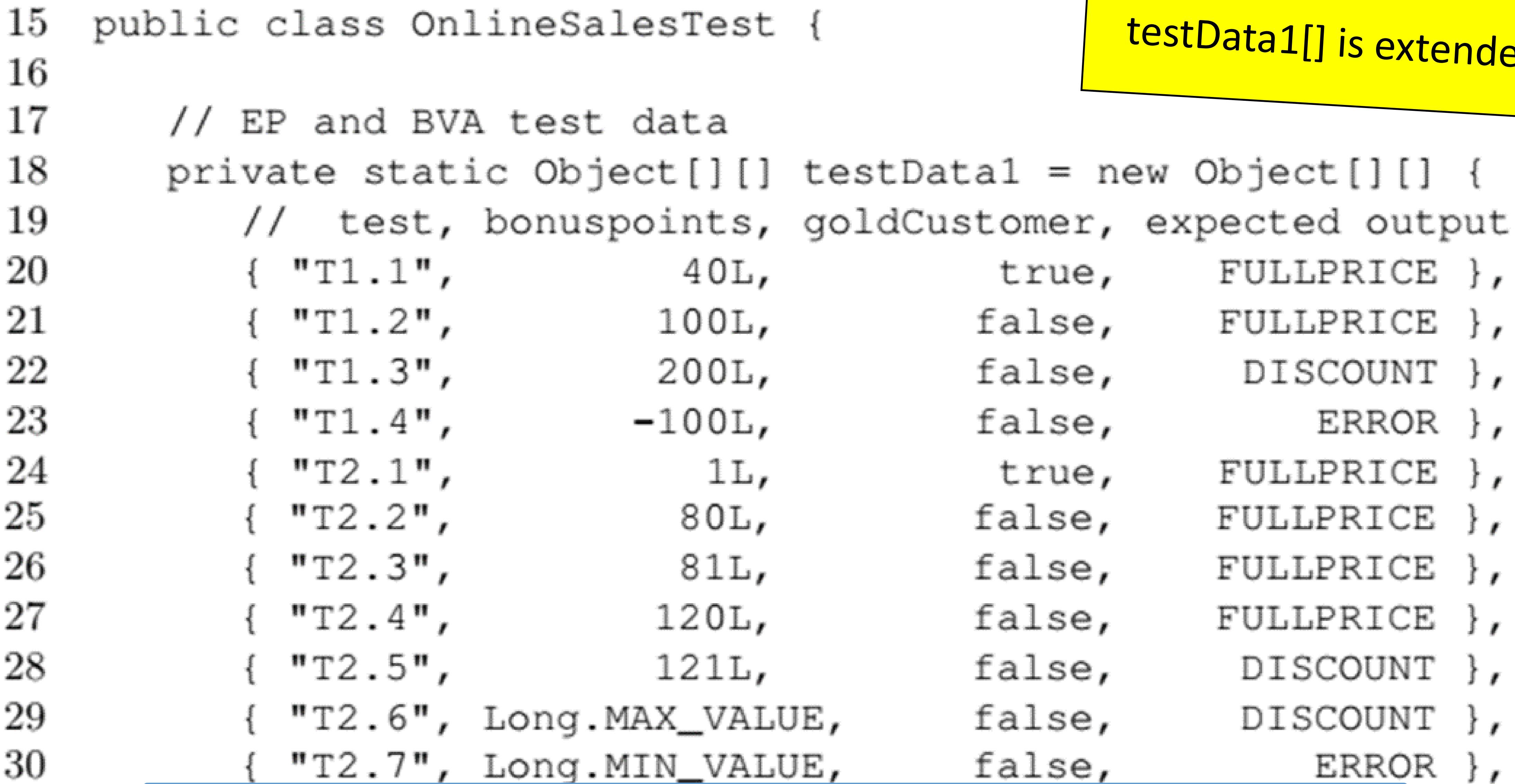
```
public class OnlineSalesTest {

    // EP and BVA test data
    private static Object[][] testData1 = new Object[][] {
       //  test, bonuspoints, goldCustomer, expected output
       { "T1.1",           40L,          true,     FULLPRICE },
       { "T1.2",          100L,         false,     FULLPRICE },
       { "T1.3",          200L,         false,      DISCOUNT },
       { "T1.4",         -100L,         false,         ERROR },
       { "T2.1",             1L,          true,     FULLPRICE },
       { "T2.2",            80L,         false,     FULLPRICE },
       { "T2.3",            81L,         false,     FULLPRICE },
       { "T2.4",           120L,         false,     FULLPRICE },
       { "T2.5",           121L,         false,      DISCOUNT },
       { "T2.6", Long.MAX_VALUE,         false,      DISCOUNT },
```



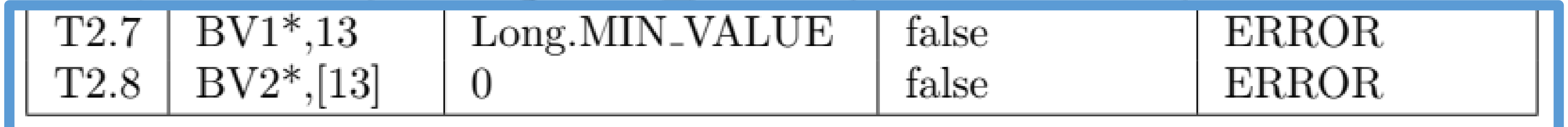
T2.6	BV8,[10,12]	Long.MAX_VALUE	false	DISCOUNT
T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

OnlineSalesTest with BVA Coverage

testData1[] is extended



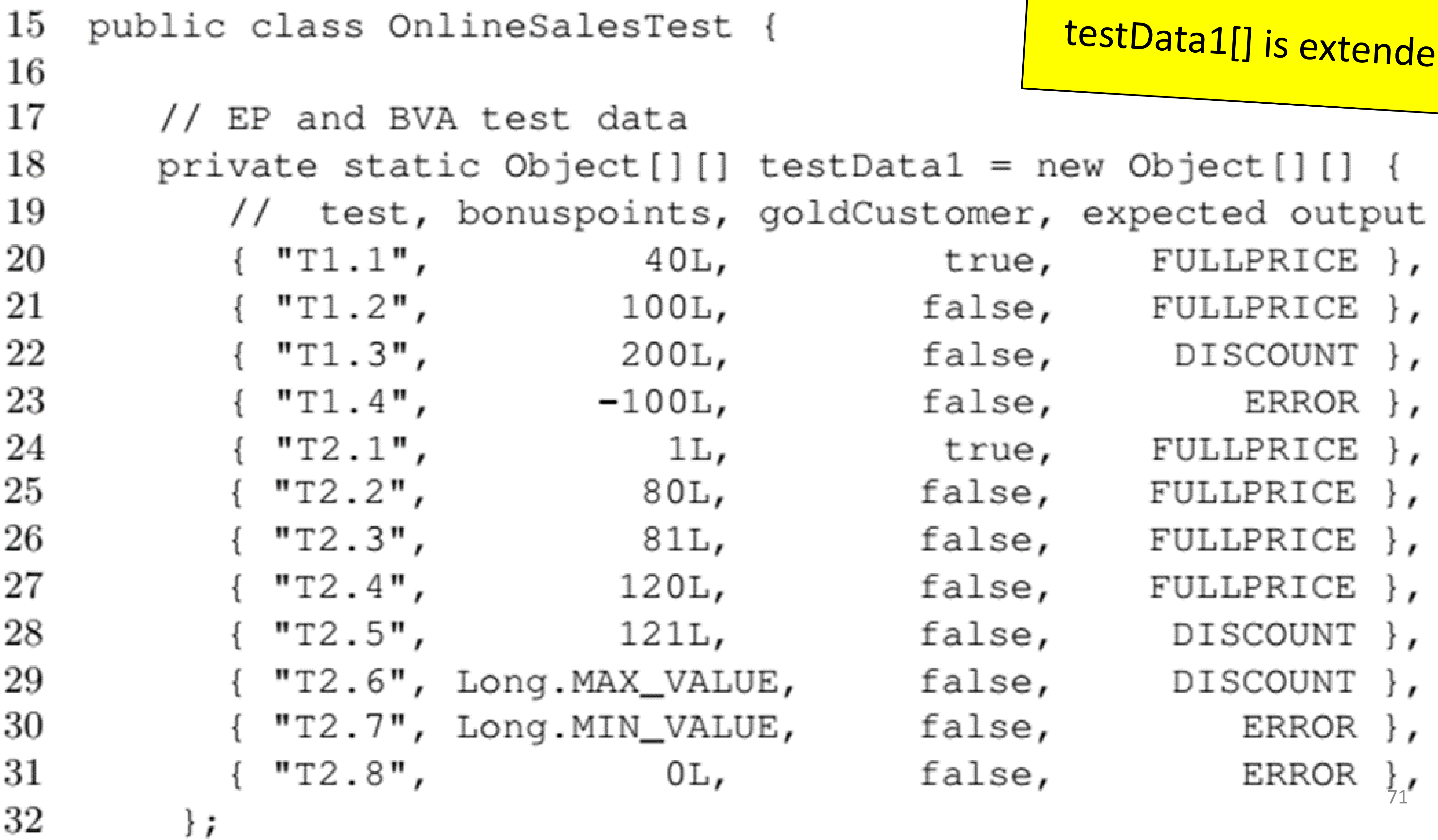
```
public class OnlineSalesTest {

    // EP and BVA test data
    private static Object[][] testData1 = new Object[][] {
       //  test, bonuspoints, goldCustomer, expected output
       { "T1.1",          40L,          true,    FULLPRICE },
       { "T1.2",         100L,         false,    FULLPRICE },
       { "T1.3",         200L,         false,     DISCOUNT },
       { "T1.4",        -100L,         false,        ERROR },
       { "T2.1",           1L,          true,    FULLPRICE },
       { "T2.2",          80L,         false,    FULLPRICE },
       { "T2.3",          81L,         false,    FULLPRICE },
       { "T2.4",         120L,         false,    FULLPRICE },
       { "T2.5",         121L,         false,     DISCOUNT },
       { "T2.6", Long.MAX_VALUE,       false,     DISCOUNT },
       { "T2.7", Long.MIN_VALUE,       false,        ERROR },
```



T2.7	BV1*,13	Long.MIN_VALUE	false	ERROR
T2.8	BV2*,[13]	0	false	ERROR

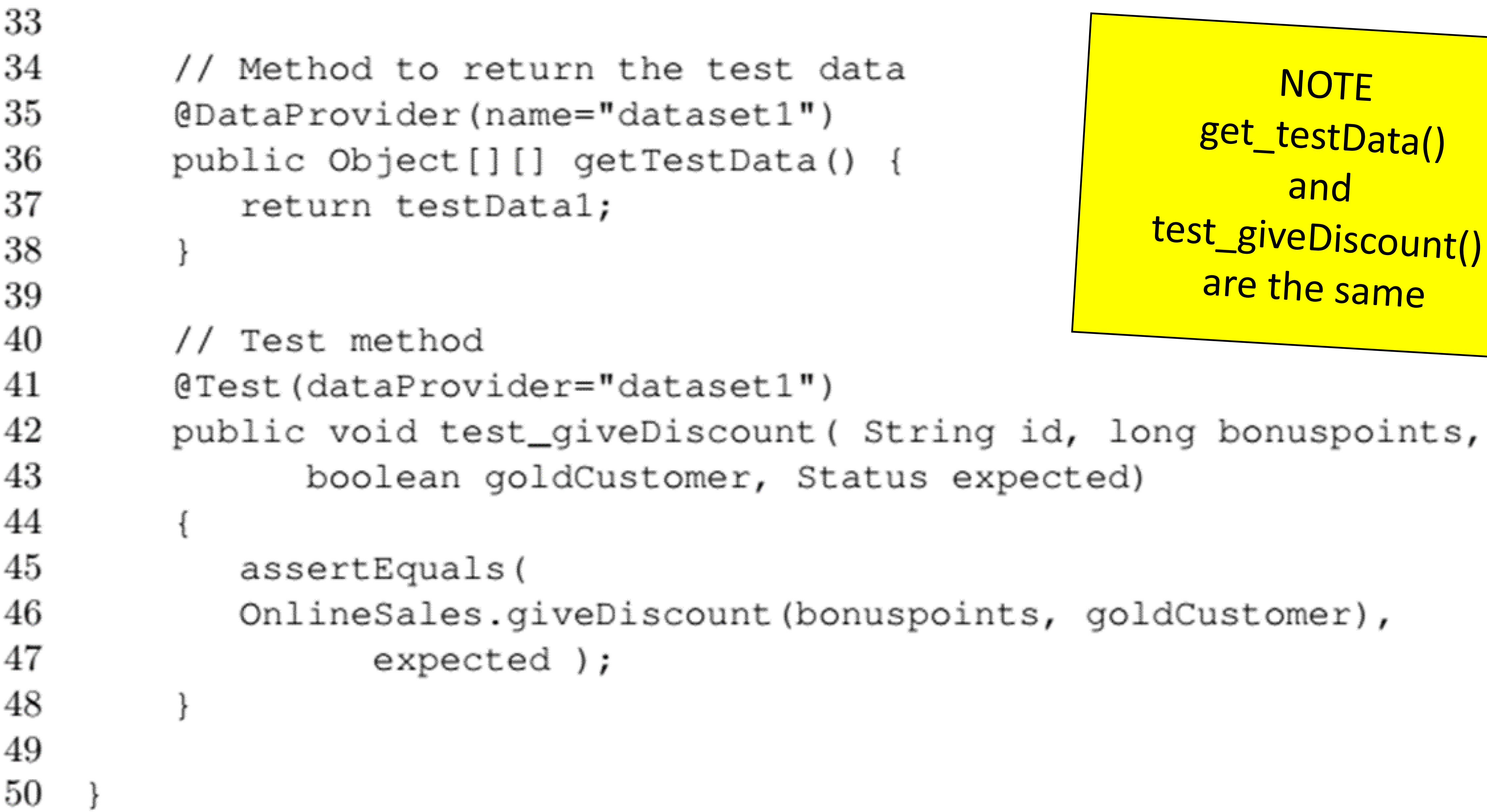
OnlineSalesTest with BVA Coverage

testData1[] is extended



```
public class OnlineSalesTest {

   // EP and BVA test data
   private static Object[][] testData1 = new Object[][] {
      //  test, bonuspoints, goldCustomer, expected output
      { "T1.1",          40L,         true,    FULLPRICE },
      { "T1.2",         100L,        false,    FULLPRICE },
      { "T1.3",         200L,        false,     DISCOUNT },
      { "T1.4",        -100L,        false,        ERROR },
      { "T2.1",           1L,         true,    FULLPRICE },
      { "T2.2",          80L,        false,    FULLPRICE },
      { "T2.3",          81L,        false,    FULLPRICE },
      { "T2.4",         120L,        false,    FULLPRICE },
      { "T2.5",         121L,        false,     DISCOUNT },
      { "T2.6", Long.MAX_VALUE,      false,     DISCOUNT },
      { "T2.7", Long.MIN_VALUE,      false,        ERROR },
      { "T2.8",           0L,        false,        ERROR },
   };
```

OnlineSalesTest with BVA Coverage



```
        // Method to return the test data
        @DataProvider(name="dataset1")
        public Object[][] getTestData() {
            return testData1;
        }

        // Test method
        @Test(dataProvider="dataset1")
        public void test_giveDiscount( String id, long bonuspoints,
                boolean goldCustomer, Status expected)
        {
            assertEquals(
            OnlineSales.giveDiscount(bonuspoints, goldCustomer),
                    expected );
        }

    }
```

NOTE
get_testData()
and
test_giveDiscount()
are the same

6. Test Execution

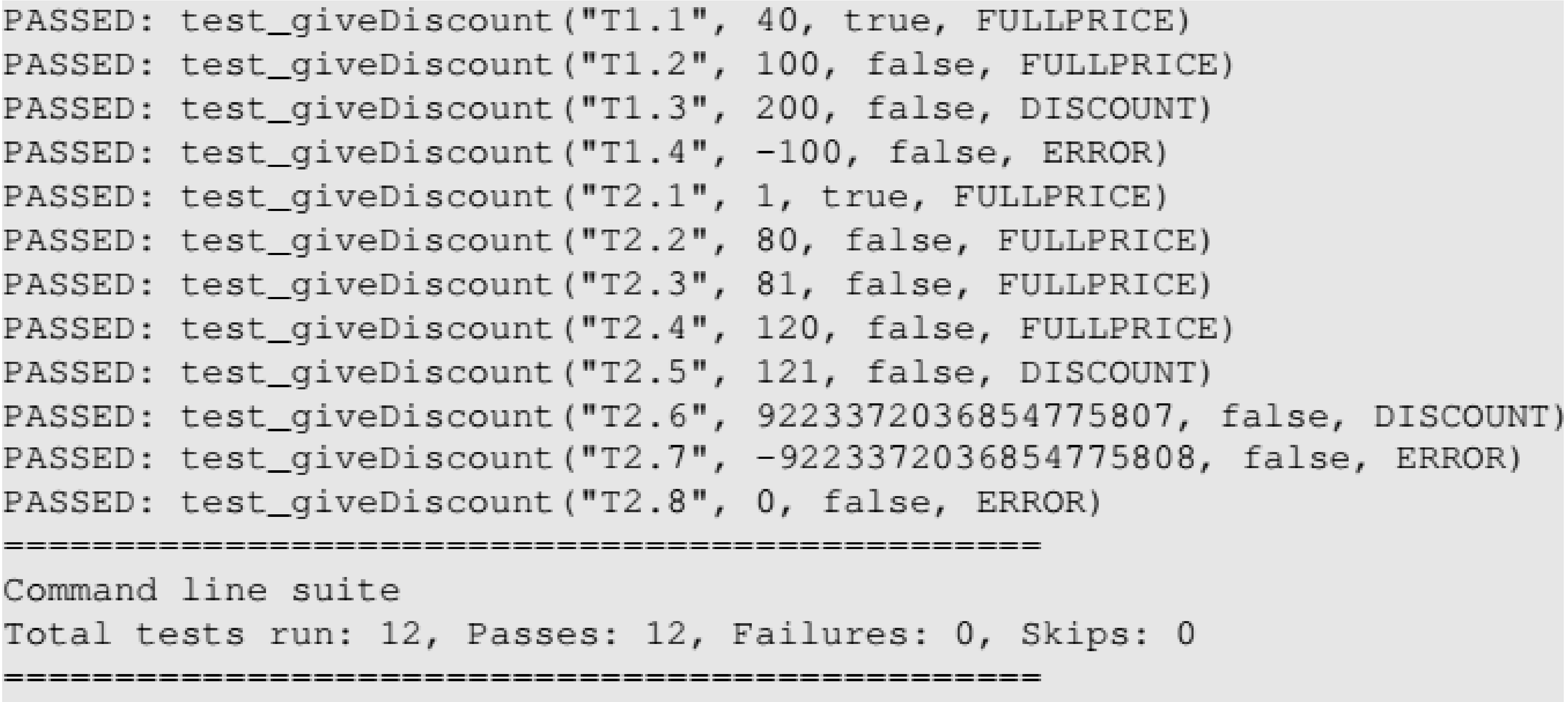
DEMO

- Running these tests against the class OnlineSales produces:

6. Test Execution

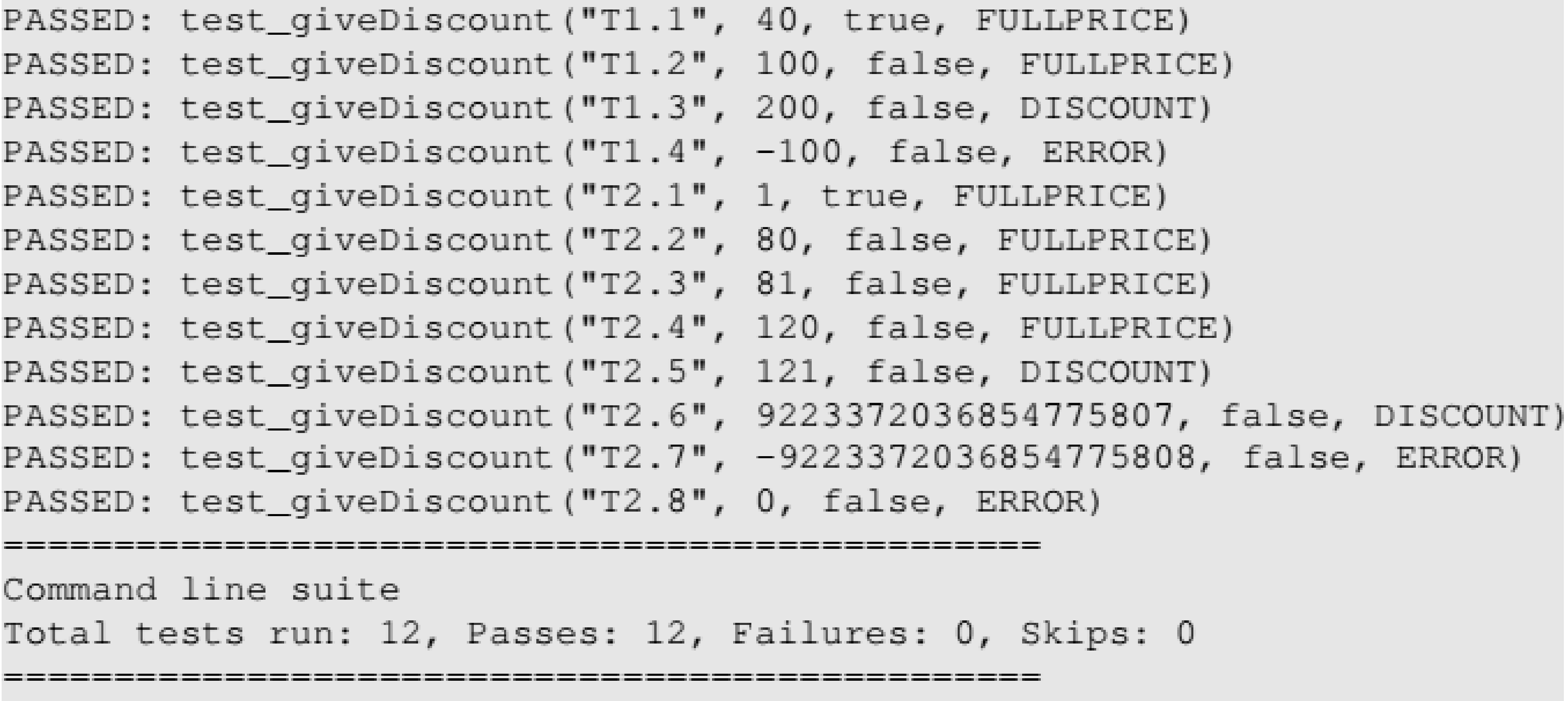
DEMO

- Running these tests against the class OnlineSales produces:



```
PASSED: test_giveDiscount("T1.1", 40, true, FULLPRICE)
PASSED: test_giveDiscount("T1.2", 100, false, FULLPRICE)
PASSED: test_giveDiscount("T1.3", 200, false, DISCOUNT)
PASSED: test_giveDiscount("T1.4", -100, false, ERROR)
PASSED: test_giveDiscount("T2.1", 1, true, FULLPRICE)
PASSED: test_giveDiscount("T2.2", 80, false, FULLPRICE)
PASSED: test_giveDiscount("T2.3", 81, false, FULLPRICE)
PASSED: test_giveDiscount("T2.4", 120, false, FULLPRICE)
PASSED: test_giveDiscount("T2.5", 121, false, DISCOUNT)
PASSED: test_giveDiscount("T2.6", 9223372036854775807, false, DISCOUNT)
PASSED: test_giveDiscount("T2.7", -9223372036854775808, false, ERROR)
PASSED: test_giveDiscount("T2.8", 0, false, ERROR)
===============================================
Command line suite
Total tests run: 12, Passes: 12, Failures: 0, Skips: 0
===============================================
```

7. Test Results



```
PASSED: test_giveDiscount("T1.1", 40, true, FULLPRICE)
PASSED: test_giveDiscount("T1.2", 100, false, FULLPRICE)
PASSED: test_giveDiscount("T1.3", 200, false, DISCOUNT)
PASSED: test_giveDiscount("T1.4", -100, false, ERROR)
PASSED: test_giveDiscount("T2.1", 1, true, FULLPRICE)
PASSED: test_giveDiscount("T2.2", 80, false, FULLPRICE)
PASSED: test_giveDiscount("T2.3", 81, false, FULLPRICE)
PASSED: test_giveDiscount("T2.4", 120, false, FULLPRICE)
PASSED: test_giveDiscount("T2.5", 121, false, DISCOUNT)
PASSED: test_giveDiscount("T2.6", 9223372036854775807, false, DISCOUNT)
PASSED: test_giveDiscount("T2.7", -9223372036854775808, false, ERROR)
PASSED: test_giveDiscount("T2.8", 0, false, ERROR)
===============================================
Command line suite
Total tests run: 12, Passes: 12, Failures: 0, Skips: 0
===============================================
```

- All the tests have passed

Next Week

- BVA in More Detail
  - Fault Model
  - Description, Analysis, Test Coverage Items, Test Cases
  - Pitfalls
- Evaluation
  - Limitations: injected faults into the code
  - Strengths & Weaknesses
- Key Points & Notes for Experienced Testers

This Afternoon

- Lab 3:
  - BVA: Analysis, Test Coverage Items, Test Cases, Review, Implement
  - Should be quick: based on lab 1 results
  - Deadline: next Monday evening (but try and get it done today)
- Submit via Moodle/quiz: Analysis, TCI, TC, review
  - Work the problems on paper first (better than using word/excel)
  - Read the instructions carefully: **ordering** very important for automated assessment

Independent Study

- Read Chapter 2: EP in more detail
- Read chapter 3