Black Box Testing - Two Examples

Flight Insurance Program Grade

Flight Insurance Example

Specification

2

A program ComputeInsurance is used by an airline company to utomatically assess the level of insurance the customer must pay on their ticket. Each customer can bring one piece of sports equipment and one piece of musical equipment on a flight:

If they bring both sports and music equipment the insurance is \in 20 If they only bring one piece of equipment only then the insurance is \in 10

If they bring no equipment then the insurance fee is $\ensuremath{\mathfrak{e}} 5$

The program input consists of two boolean variables: sportsEquipment musicEquipment

The program output is a single variable: insurance

1

Input Partitions

Parameter	Test Case	Partition Range
sportsEquipment	1	True
	2	False
musicEquipment	3	True
	4	False

Output Partitions

Parameter	Test Case	Partition Range
Insurance	5	5
	6	10
	7	20

3

Test Cases

· Each Partition is as Test Case

Test Data

Test No.	Test Cases/ Equivalence Partitions Covered	In	puts	Expected Outputs
		sportsEquipment	musicEquipment	Insurance
1	1, 3, 7	True	True	20
2	1, 4, 6	True	False	10
3	2, 4, 5	False	False	5

5

Boundary Value Analysis

 Testing using Boundary Value Analysis is not useful with boolean variables as their values can only be True or False. It is thus much more suited to numerical variables.

Combinational Testing

The Causes for the program taken from the specification are

Causes

sportsEquipment=True; sportsEquipment=False; musicEquipment=True; musicEquipment=False;

They effects they generate are

Effects

8

10

insurance=5; insurance=10; insurance=20;

7

Test Data

Test No.	Test Cases/ Rules Covered	Inputs		Expected Outputs
		sportsEquipment	musicEquipment	Insurance
1	1	False	False	5
2	2	True	False	10
3	3	False	True	10
4	4	True	True	20

Truth Table

			Rules	
	1	2	3	4
Causes				
sportsEquipment	F	Т	F	Т
musicEquipment	F	F	Т	Т
Effects				
Insurance=5	Т	F	F	F
Insurance=10	F	Т	Т	F
Insurance=20	F	F	F	Т

9

Program Grade

- Specification
 - The program 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 over 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'.

Equivalence Partitioning

· Input Partitions

Parameter	Test Case	Partition Range
Exam	1*	INT_MIN1
	2	049
	3	5059
	4	60100
	5*	101 INT_MAX
Course	6*	INT_MIN1
	7	049
	8	50100
	Q#	101 INT MAX

11 12

Equivalence Partitions

· Output Partitions

Parameter	Test Case	Partition Range
Grade	10	'Marks out of range'
	11	'Fail'
	12	'Pass,C'
	13	'Pass,B'
	14	'Pass,A'

Each Partition is a Test Case

Test No.	Test Cases/Partitions	Inputs		Expected Outputs
	Covered	exam	course	Result
1	2, 7, 11	40	40	Fail
4	3, 8, 12	55	50	Pass, C
5	4, 8, 13	80	50	Pass, B
6	4, 8, 14	90	50	Pass, A
7*	1	-1	50	Marks out of Range
8*	5	101	50	Marks out of range
9*	6	50	-1	Marks out of range
10*	9	50	101	Marks out of range

13 14

Boundary Value Analysis

Input Boundary Values

Parameter	Test Case	Partition Boundary value
Exam	1*	INT_MIN
	2*	-1
	3	0
	4	49
	5	50
	6	59
	7	60
	8	100
	9*	101
	10*	INT_MAX
Course	11*	INT MIN
	12*	-1
	13	0
	14	49
	15	50
	16	100
	17*	101
	18*	INT_MIN

Boundary Value Analysis

Output Boundary Values

16

Parameter	Test Case	Partition Boundary valu
Grade	19	'Marks out of range'
	20	'Fail'
	21	'Pass,C'
	22	'Pass,B'
	23	'Pass,A'

15

Each Boundary Value is a Test Case

Test No.	Test Cases/Boundaries	Inputs		Expected Outputs
	Covered	exam	course	Result
1	3, 15, 20	0	50	Fail
2	4, 15, 20	49	50	Fail
3	5, 15, 21	50	50	Pass, C
4	6, 15, 21	59	50	Pass, C
5	7, 15, 22	60	50	Pass, B
6	8, 15, 23	100	50	Pass, A
7	5, 13, 20	50	0	Fail
8	5,14, 20	50	49	Fail
9	5, 15, 21	50	50	Pass, C

Each Boundary Value is a Test Case

10	5, 16, 23	50	100	Pass, A
11*	1, 15, 19	INT_MIN	50	Marks out of
				Range
12*	2, 15, 19	-1	50	Marks out of
				Range
13*	9, 15, 19	101	50	Marks out of
				range
14*	10, 15, 19	INT_MAX	50	Marks out of
				range
15*	5, 11, 19	50	INT_MIN	Marks out of
				Range
16*	5, 12, 19	50	-1	Marks out of
				range
17*	5, 17, 19	50	101	Marks out of
				range
18*	5, 18, 19	50	INT_MAX	Marks out of
				Range

Combinational Testing

								_	
	1	2	3	4	5	6	7	8	9
Causes									
Exam<0	T	*	F	*	F	F	F	F	F
Exam>100	F	*	T	*	F	F	F	F	F
course<0	*	T	*	*	F	F	F	F	F
course>100	*	F	*	T	F	F	F	F	F
0<=exam<50	F	*	F	*	T	*	F	F	F
0<=course<50	*	F	*	F	*	T	F	F	F
50<=exam<60	F	*	F	*	F	*	T	F	F
60<=exam<=100	F	*	F	*	F	*	F	T	T
50<=course<=100	*	F	*	F	*	F	T	T	T
(exam+course)./2>=70	*	*	*	*	*	*	*	F	T
Effects									
Marks out of range	T	T	T	T	F	F	F	F	F
Fail	F	F	F	F	T	T	F	F	F
Pass, C	F	F	F	F	F	F	T	F	F
Pass,B	F	F	F	F	F	F	F	T	F
Pass,A	F	F	F	F	F	F	F	F	T

Each Rule is a test case

Test Data

Test No.	Test Cases/Rules Covered	Inputs	Expected Outputs	
		Exam	course	Result
1	5	40	50	Fail
2	6	50	40	Fail
3	7	55	50	Pass, C
4	8	80	50	Pass, B
5	9	90	50	Pass, A
6*	1	-1	50	Marks out of Range
7*	2	50	-1	Marks out of range
8*	3	101	50	Marks out of range
9*	4	50	101	Marks out of range