



IIC3745 - Testing (2020 / II)

Actividad 2 - Solución

Para todos los criterios de cobertura, tenemos los siguientes predicados y cláusulas:

Predicados

p1	@is_hardcover
p2	@is_hardcover and @is_frecuent_client
p3	@is_hardcover and !@is_frecuent_client
p4	@total_pages > 500 and total_pages_images >= 6
p5	@is_premium_book and (@pages_of_text > 300 or @total_pages_images >= 5) and !@is_frecuent_client

Cláusulas

c1	@is_hardcover
c2	@is_frecuent_client
c3	@total_pages > 500
c4	@total_pages_images >= 6
c5	@is_premium_book
c6	@pages_of_text > 300
c7	@total_pages_images >= 5

Cada criterio de cobertura tiene distintos requerimientos y casos de prueba que se detallan en esta pauta. Además, a modo de complementar, cada caso de prueba en la solución tiene una implementación de código en [este repositorio](#) bajo el mismo formato pedido para la entrega.

1. Clause Coverage / Conditional Coverage

Tanto para *clause* como para *conditional* coverage, los requisitos de prueba son los mismos - que cada cláusula evalúe tanto **true** como **false**. Los requerimientos se muestran en formato tabla a continuación:

Test Requirements (TRs)

	c1	c2	c3	c4	c5	c6	c7
TR1	true	-	-	-	-	-	-
TR2	false	-	-	-	-	-	-
TR3	-	true	-	-	-	-	-
TR4	-	false	-	-	-	-	-
TR5	-	-	true	-	-	-	-
TR6	-	-	false	-	-	-	-
TR7	-	-	-	true	-	-	-
TR8	-	-	-	false	-	-	-
TR9	-	-	-	-	true	-	-
TR10	-	-	-	-	false	-	-
TR11	-	-	-	-	-	true	-
TR12	-	-	-	-	-	false	-
TR13	-	-	-	-	-	-	true
TR14	-	-	-	-	-	-	false

Por la discusión en la issue [#35](#) del *syllabus* del curso, se considerará como correcta cualquiera de las dos siguientes resoluciones respecto a los casos de prueba.

1.1 Clause coverage

Este criterio no toma en consideración los cortocircuitos que pueden generar los predicados en la evaluación efectiva de la cláusula. En base a esto y los 14 requisitos de prueba, tenemos entonces dos posibles casos de prueba:

Test cases (TCs)

	c1	c2	c3	c4	c5	c6	c7	Output
TC1	true	true	true	true	true	true	true	10000
TC2	false	false	false	false	false	false	false	4000

A continuación se presentan los *inputs* y *outputs* para ejecutarlos:

	Pages	Double pages	Images	Hardcover	Premium book	Frequent Client	Output
TC1	600	5	5	true	true	true	10000
TC2	100	0	0	false	false	false	4000

Esta aproximación es bastante trivial y no aporta mucho valor en la prueba real de lógica del código, pero dada la ambigüedad de las explicaciones y diapositivas, se considera correcto.

1.2. Conditional Coverage

Este criterio tiene la particularidad de que los casos de prueba deben ser capaces de efectivamente evaluar el requerimiento de cada cláusula. A modo de ejemplo, para el predicado **a and b**:

a and b	a	b
TC1	true	true
TC2	false	false

en verde los casos que son evaluados por primera vez
en gris los casos que no son evaluados por cortocircuito de predicado

El conjunto de casos de prueba no satisface el criterio de cobertura. Esto porque TC2 no evalúa efectivamente que **b = true** dado que **a = false** (el **and** no continua la evaluación del segundo término si el primero es **false**). En este sentido, los predicados del ejercicio original escritos en base a las cláusulas respectivas son:

p1	c1
p2	c1 and c2
p3	c1 and !c2
p4	c3 and c4
p5	c5 and (c6 or c7) and !c2

De aquí podemos ver que c1 puede prevenir la evaluación de c2, c3 puede prevenir la evaluación de c4 y alguna combinación de c5, c6 y c7 pueden también evitar la evaluación de c2. Con esto en consideración, los casos de prueba para este criterio de cobertura serían:

Test cases (TCs)

	c1	c2	c3	c4	c5	c6	c7
TC1	true	true	true	true	true	true	true
TC2	false	false	true	false	true	false	false
TC3	true	false	false	false	false	false	false

en verde los casos que son evaluados por primera vez
en gris los casos que ya fueron evaluados o no son evaluados por cortocircuito de predicado

Los *inputs* y *outputs* para lograr estos casos de prueba se detallan a continuación:

	Pages	Double pages	Images	Hardcover	Premium book	Frequent Client	Output
TC1	600	5	5	true	true	true	10000
TC2	100	0	0	false	true	false	4000
TC3	100	0	0	true	false	false	8000

Este criterio es un poco más riguroso y mejor en cuanto a análisis de lógica que el anterior, pues toma en consideración la forma del predicado para generar los casos de prueba asociados. Era originalmente al que nos referíamos en el enunciado.

2. Correlated Active Clause Coverage

Se debe realizar el análisis para cada predicado

p1	c1
p2	c1 and c2
p3	c1 and !c2
p4	c3 and c4
p5	c5 and (c6 or c7) and !c2

Notemos que c3 y c6 son dependientes entre sí, la combinación imposible es que c3 = true and c6 = false (páginas no puede ser mayor a 600 y menor a 300 a la vez). Lo mismo sucede con c4 y c7, la combinación imposible es que c4 = true and c7 = false (páginas con imágenes no puede ser mayor o igual a 6 y menor o igual a 5 a la vez). Estas restricciones se deben considerar en el diseño de pruebas.

Predicado P1

Solo se tiene una cláusula, por lo que esta es activa y quedan dos combinaciones posibles:

p1/c1	c1	c2	c3	c4	c5	c6	c7
CS1	true	-	-	-	-	-	-
CS2	false	-	-	-	-	-	-

Predicado P2

c1 es cláusula activa: se debe dar que c2 = true. Se desprenden dos combinaciones posibles:

p2/c1	c1	c2	c3	c4	c5	c6	c7
CS3	true	true	-	-	-	-	-
CS4	false	true	-	-	-	-	-

c2 es cláusula activa: se debe dar que $c1 = \text{true}$. Se desprenden dos combinaciones posibles:

p2/c2	c1	c2	c3	c4	c5	c6	c7
CS5	true	true	-	-	-	-	-
CS6	true	false	-	-	-	-	-

Predicado P3

c1 es cláusula activa: se debe dar que $c2 = \text{false}$. Se desprenden dos combinaciones posibles:

p3/c1	c1	c2	c3	c4	c5	c6	c7
CS7	true	false	-	-	-	-	-
CS8	false	false	-	-	-	-	-

c2 es cláusula activa: se debe dar que $c1 = \text{true}$. Se desprenden dos combinaciones posibles:

p3/c2	c1	c2	c3	c4	c5	c6	c7
CS9	true	true	-	-	-	-	-
CS10	true	false	-	-	-	-	-

Predicado P4

c3 es cláusula activa: se debe dar que $c4 = \text{true}$. Se desprenden dos combinaciones posibles:

p4/c3	c1	c2	c3	c4	c5	c6	c7
CS11	-	-	true	true	-	-	-
CS12	-	-	false	true	-	-	-

c4 es cláusula activa: se debe dar que $c3 = \text{true}$. Se desprenden dos combinaciones posibles:

p3/c4	c1	c2	c3	c4	c5	c6	c7
CS13	-	-	true	true	-	-	-
CS14	-	-	true	false	-	-	-

Predicado P5

c2 es cláusula activa: se debe dar que $c5 = \text{true}$ y que $c6 = \text{true}$ o $c7 = \text{true}$, De esto se desprenden cuatro combinaciones posibles:

p5/c2	c1	c2	c3	c4	c5	c6	c7
CS15	-	false	-	-	true	true	false
CS16	-	true	-	-	true	true	false
CS17	-	false	-	-	true	false	true
CS18	-	true	-	-	true	false	true

De acá solamente necesitamos dos combinaciones por definición de CACC. Dado que no hay otros tests que involucren $c5$, $c6$ y $c7$, podemos escoger arbitrariamente CS15 y CS16 sin ser redundantes.

c5 es cláusula activa: se debe dar que $c2 = \text{false}$ y que $c6 = \text{true}$ o $c7 = \text{true}$, De esto se desprenden cuatro combinaciones posibles:

p5/c5	c1	c2	c3	c4	c5	c6	c7
CS19	-	false	-	-	true	true	false
CS20	-	false	-	-	false	true	false
CS21	-	false	-	-	true	false	true
CS22	-	false	-	-	false	false	true

Análogo al caso anterior, se escogen dos combinaciones de las cuatro posibles: CS19 y CS20.

c6 es cláusula activa: se debe dar que $c2 = \text{false}$, $c5 = \text{true}$ y que $c7 = \text{false}$. Se desprenden dos combinaciones posibles:

p5/c6	c1	c2	c3	c4	c5	c6	c7
CS23	-	false	-	-	true	true	false
CS24	-	false	-	-	true	false	false

c7 es cláusula activa: se debe dar que $c2 = \text{false}$, $c5 = \text{true}$ y que $c6 = \text{false}$. Se desprenden dos combinaciones posibles:

p5/c7	c1	c2	c3	c4	c5	c6	c7
CS25	-	false	-	-	true	false	true
CS26	-	false	-	-	true	false	false

De todos los predicados, tenemos la siguiente tabla de combinaciones de prueba:

	c1	c2	c3	c4	c5	c6	c7
CS1	true	-	-	-	-	-	-
CS2	false	-	-	-	-	-	-
CS3	true	true	-	-	-	-	-
CS4	false	true	-	-	-	-	-
CS5	true	true	-	-	-	-	-
CS6	true	false	-	-	-	-	-
CS7	true	false	-	-	-	-	-
CS8	false	false	-	-	-	-	-
CS9	true	true	-	-	-	-	-
CS10	true	false	-	-	-	-	-
CS11	-	-	true	true	-	-	-

CS12	-	-	false	true	-	-	-
CS13	-	-	true	true	-	-	-
CS14	-	-	true	false	-	-	-
CS15	-	false	-	-	true	true	false
CS16	-	true	-	-	true	true	false
CS19	-	false	-	-	true	true	false
CS20	-	false	-	-	false	true	false
CS23	-	false	-	-	true	true	false
CS24	-	false	-	-	true	false	false
CS25	-	false	-	-	true	false	true
CS26	-	false	-	-	true	false	false

Notemos que hay varias que se repiten. Al fusionarlas, tenemos nuestra tabla de requisitos de prueba

Test Requirements (TRs)

	c1	c2	c3	c4	c5	c6	c7
TR1	true	-	-	-	-	-	-
TR2	false	-	-	-	-	-	-
TR3	true	true	-	-	-	-	-
TR4	false	true	-	-	-	-	-
TR5	true	false	-	-	-	-	-
TR6	false	false	-	-	-	-	-
TR7	-	-	true	true	-	-	-
TR8	-	-	false	true	-	-	-
TR9	-	-	true	false	-	-	-

TR10	-	false	-	-	true	true	false
TR11	-	true	-	-	true	true	false
TR12	-	false	-	-	false	true	false
TR13	-	false	-	-	true	false	false
TR14	-	false	-	-	true	false	true

Luego, podemos desprender una posible combinación de casos de prueba

Test cases (TCs)

	TRs	c1	c2	c3	c4	c5	c6	c7
TC1	1, 5, 8, 14	true	false	false	true	true	false	true
TC2	2, 6, 9, 10	false	false	true	false	true	true	false
TC3	3, 7	true	true	true	true	true	true	true
TC4	4, 11	false	true	true	false	true	true	false
TC5	12	false	false	false	false	false	true	false
TC6	13	false	false	false	false	true	false	false

Estos casos de prueba se pueden lograr a través de la siguiente combinación de *inputs* y *outputs*:

	Pages	Double pages	Images	Hardcover	Premium book	Frequent Client	Output
TC1	100	5	5	true	true	false	16000
TC2	600	0	0	false	true	false	8000
TC3	600	5	5	true	true	true	10000
TC4	600	0	0	false	true	true	4000
TC5	400	0	0	false	false	false	4000
TC6	100	0	0	false	true	false	4000

3. Line Coverage

Los posibles casos de prueba se pueden simplificar mucho dado que son tres controles de flujo independientes. En particular, una alternativa de casos de prueba para asegurar que cada línea sea evaluada al menos una vez se muestra a continuación:

	P1	P2	P3	P4	P5
TC1	true	false	true	true	true
TC2	true	true	false	false	false

Una posible combinación de *inputs* y *outputs* para ejecutar estos casos de prueba puede ser:

	Pages	Double pages	Images	Hardcover	Premium book	Frequent Client	Output
TC1	600	5	5	true	true	false	24000
TC2	100	0	0	true	false	true	6000

La pregunta se corregirá en base al porcentaje de coverage de línea indicado por SimpleCov y si no hay casos de prueba redundantes.

4. Branch Coverage

Para asegurar que cada predicado sea **true** y **false**, una alternativa puede ser los siguientes tres casos de prueba:

	p1	p2	p3	p4	p5
TC1	true	true	false	false	false
TC2	true	false	true	true	true
TC3	false	-	-	true	true

Un ejemplo de *inputs* y *outputs* para diseñar estos casos de prueba se detalla a continuación:

	Pages	Double pages	Images	Hardcover	Premium book	Frequent Client	Output
TC1	100	0	0	true	false	true	6000

TC2	600	5	5	true	true	false	24000
TC3	600	5	5	false	true	false	16000

La pregunta se corregirá en base al porcentaje de coverage de línea indicado por SimpleCov y si no hay casos de prueba redundantes.