**Q1. Chapter 2, Section 2.4, Question 3 (pp. 74-75 43 of your textbook).**



1. Call sites: line 12.

trash() -> takeOut()

1. All pairs of last-def and first-uses:

|  |  |  |
| --- | --- | --- |
|  | **Last-def** | **First-uses** |
| 1 | (trash(),m,5) | (takeOut(),a,19) |
| 2 | (trash(),m,7) | (takeOut(),a,19) |
| 3 | (trash(),n,9) | (takeOut(),b,21) |
| 4 | (trash(),n,9) | (takeOut(),b,23) |
| 5 | (trash(),n,11) | (takeOut(),b,21) |
| 6 | (trash(),n,11) | (takeOut(),b,23) |
| 7 | (takeOut(),e,21) | (trash(),o,13) |
| 8 | (takeOut(),e,23) | (trash(),o,13) |

1. Test input x <=0 (for example, x=0)satisfies TR 1, 6, 8.

Test input x > 5 (for example, x=6) satisfies TR 2, 3, 7.

Test input 1<=x <=5 (for example=3) satisfies TR 2,5,7.

TR 4 can not be satisfied because if x>5 , m=4 and n=12 , calling takeout(4,12) , making **a** in takeOut always >0 – forcing line 23 to never be called for last-def ((trash(),n,9)).

**Q2. Chapter 2, Section 2.5, Question 2 (page 87 of your textbook).**

1. 4 states:  
   A = {Wake,Low}

B = {Wake, High}

C = {Sleep, Low}

D = {Sleep, High}

C:\Users\skshimon\Dropbox\Spring 2016\CS580 A5\Homework\a4\Q2_A4.png

1. Edge coverage TR =

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | (A,A) | 5 | (B,D) | 9 | (C,D) |
| 2 | (A,B) | 6 | (D,B) | 10 | (C,C) |
| 3 | (B,A) | 7 | (D,D) | 11 | (C,A) |
| 4 | (B,B) | 8 | (D,C) | 12 | (A,C) |

Test case = {2,4,5,7,6,3,1,12,9,8,10,11} satisfies edge coverage on the FSM. The sequence of calls from state A (Wake, Low) is

up() 🡪up()🡪advance()🡪up()🡪advance()🡪down() 🡪down() 🡪advance() 🡪up() 🡪down() 🡪down() 🡪advance() .

This sequence of calls ensure that the thermostat returns to the same state (before the test) after the test sequence ends.

**Q3. Chapter 3, Section 3.2; do parts (a)-(h) for the predicate in Question 7 (page 119 of your textbook).**

***p* = (*a* ∨ *b*) ∧ (*c* ∨ *d*)**

1. Clauses: a, b, c, d
2. pa = pa=true ⊕ pa=false

= (true ∨ *b*) ∧ (*c* ∨ *d*) ⊕ (false ∨ *b*) ∧ (*c* ∨ *d*)

= (true) ∧ (*c* ∨ *d*) ⊕ (*b*) ∧ (*c* ∨ *d*)

= (*c* ∨ *d*) ⊕ (*b*) ∧ (*c* ∨ *d*)

**= ¬*b* ∧ (*c* ∨ *d*)**

pb = pb=true ⊕ pb=false

= (*a* ∨ *true*) ∧ (*c* ∨ *d*) ⊕ (*a ∨* *false*) ∧ (*c* ∨ *d*)

= (true) ∧ (*c* ∨ *d*) ⊕ (*a*) ∧ (*c* ∨ *d*)

= (*c* ∨ *d*) ⊕ (*a*) ∧ (*c* ∨ *d*)

**= ¬*a* ∧ (*c* ∨ *d*)**

pc = pc=true ⊕ pc=false

= (a ∨ *b*) ∧ (*true* ∨ *d*) ⊕ (a ∨ *b*) ∧ (*false* ∨ *d*)

= (a ∨ *b*) ∧ (*true*) ⊕ (a ∨ *b*) ∧ (*d*)

= (a ∨ *b*) ⊕ (a ∨ *b*) ∧ (*d*)

**= ¬*d* ∧ (*a* ∨ *b*)**

pd = pd=true ⊕ pd=false

= (a ∨ *b*) ∧ (*c* ∨ *true*) ⊕ (a ∨ *b*) ∧ (*c* ∨ *false*)

= (a ∨ *b*) ∧ (*true*) ⊕ (a ∨ *b*) ∧ (*c*)

= (a ∨ *b*) ⊕ (a ∨ *b*) ∧ (*c*)

**= ¬*c* ∧ (*a* ∨ *b*)**

1. Truth Table:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | a | b | c | d | pa**= ¬*b*∧(*c*∨*d*)** | pb**= ¬*a*∧(*c*∨*d*)** | pc**= ¬*d*∧(*a*∨*b*)** | pd**=**  **¬*c*∧(*a*∨ *b*)** | ***p* =**  **(*a* ∨ *b*) ∧ (*c* ∨ *d*)** |
| 1 | T | T | T | T | F | F | F | F | T |
| 2 | T | T | T | F | F | F | T | F | T |
| 3 | T | T | F | T | F | F | F | T | T |
| 4 | T | T | F | F | F | F | T | T | F |
| 5 | T | F | T | T | T | F | F | F | T |
| 6 | T | F | T | F | T | F | T | F | T |
| 7 | T | F | F | T | T | F | F | T | T |
| 8 | T | F | F | F | F | F | T | T | F |
| 9 | F | T | T | T | F | T | F | F | T |
| 10 | F | T | T | F | F | T | T | F | T |
| 11 | F | T | F | T | F | T | F | T | T |
| 12 | F | T | F | F | F | F | T | T | F |
| 13 | F | F | T | T | T | T | F | F | F |
| 14 | F | F | T | F | T | T | F | F | F |
| 15 | F | F | F | T | T | T | F | F | F |
| 16 | F | F | F | F | F | F | F | F | F |

1. With respect to clause a, GACC pairs are = {5,6,7}x{13,14,15}

With respect to clause b, GACC pairs are = {9,10,11}x{13,14,15}

With respect to clause c, GACC pairs are = {2,6,10}x{4,8,12}

With respect to clause d, GACC pairs are = {3,7,11}x{4,8,12}

1. With respect to clause a, CACC pairs are = {5,6,7}x{13,14,15}

With respect to clause b, CACC pairs are = {9,10,11}x{13,14,15}

With respect to clause c, CACC pairs are = {2,6,10}x{4,8,12}

With respect to clause d, CACC pairs are = {3,7,11}x{4,8,12}

1. With respect to clause a, RACC pairs are = (5,13),(6,14),(7,15)

With respect to clause b, RACC pairs are = (9,13),(10,14),(11,15)

With respect to clause c, RACC pairs are = (2,4),(6,8),(10,12)

With respect to clause d, RACC pairs are = (3,4),(7,8),(11,12)

1. With respect to clause A, GICC 4-tuples are = {1,2,3}x{9,10,11}x{4,8}x{12,16}

With respect to clause B, GICC 4-tuples are = {1,2,3}x{5,6,7}x{4,12}x{8,16}

With respect to clause C, GICC 4-tuples are = {1,5,9} x{3,7,11}x{13,14} x{15,16}

With respect to clause D, GICC 4-tuples are = {1,5,9}x{2,6,10}x{13,15}x{14,16}

1. With respect to clause A, RICC 4-tuples are = {(1,9),(2,10),(3,11)}**x**{(4,12),(8,16)}

With respect to clause B, RICC 4-tuples are = {(1,5),(2,6),(3,7)}**x**{(4,8),(12,16)}

With respect to clause C, RICC 4-tuples are = {(1,3),(5,7),(9,11)}**x**{(13,15) ,(15,16)}

With respect to clause D, RICC 4-tuples are = {(1,2),(5,6),(9,10)}**x**{(13,14) ,(15,16)}

**Q4. Chapter 3, Section 3.3, Question 2 (page 130 of your textbook).**

public String twoPred(int x, int y) //line 1

{ //line 2

boolean z; //line 3

if(x<y) //line 4

z=true; //line 5

else //line 6

z = false; //line 7

if(z && x+y==10) //line 8

return “A”; //line 9

else //line 10

return “B”; //line 11

} //line 12

From line 3 and 4, the truth value of z depends on the predicate p=(x<y) on line 4.

If x<y == true, z = true and if x<y == false, z = false.

So, predicate at line 8 can be written as p = (x<y) && (x+y==10).

Say,

(x<y) is clause **a**

And

(x+y==10) is clause **b**

**So , predicate at line 8 , p = a&&b.**

Now, **pa** (a major clause) = (true &&b) ⊕(false &&b) = b⊕false = b.

And similarly, **pb** (a major clause) = a.

So, RACC truth table for p,

|  |  |  |
| --- | --- | --- |
| a | b |  |
| T | T | a major clause |
| F | T |
| T | T | b major clause |
| T | F |

From this truth table, we can see that there are three unique clause combinations to satisfy RACC, (a,b)=(T,T),(T,F),(F,T)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a(x<y) | b(x+y==10) | possible value for x | Possible value for y | Test |
| T | T | 3 | 7 | twoPred(3,7) |
| T | F | 3 | 8 | twoPred(3,8) |
| F | T | 7 | 3 | twoPred(7,3) |

So, for **RACC , test cases are** twoPred(3,7), twoPred(3,8), twoPred(7,3).

**RICC:**

For RICC, major clauses will not determine p.

We have already seen from RACC that when a is major clause, only b = true will make this clause determine p . If b=false, a will not determine p = a&&b .

For b=false, RICC will have no feasible pair for p=true. The truth table is

|  |  |  |
| --- | --- | --- |
| **a** | **b** | **P** |
| t | f | F |
| f | f | F |

Similarly, when b is major clause, only a=true will make this clause determine p. If a = false, major clause b will be inactive.

For a=false, the truth table is

|  |  |  |
| --- | --- | --- |
| **a** | **b** | **p** |
| f | t | f |
| f | f | f |

Similar to b=false, when a=false, RICC will have no feasible pair for p=true.

From these two truth tables, we see that the combinations (a,b)={(t,f),(f,t),(f,f)} satisfy the RICC requirements.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a(x<y) | b(x+y==10) | possible value for x | Possible value for y | Test |
| t | f | 3 | 8 | twoPred(3,8) |
| f | t | 7 | 3 | twoPred(7,3) |
| f | F | 7 | 5 | twoPred(7,5) |

**So, for RICC , test cases are twoPred(7,5), twoPred(3,8), twoPred(7,3).**