**2.1(a)**

|  |
| --- |
| **Predicates** |
| **P1: s1<=0 | s2<= 0 | s3<=0** |
| **P2: s1+s2<=s3 | s2+s3<=s1 | s1+s3<=s2** |
| **P3: s1==s2 & s2 ==s3** |
| **P4: s1== s2| s2==s3 | s1==s3** |

|  |  |
| --- | --- |
| **PC TR1** | **s1<=0|s2<=0|s3<=0**  TR1 is a test requirement that makes a predicate **P1** in the condensation graph for Question 1, true, attempting to achieve predicate coverage. |
| **TR2** | **s1>0 & s2>0 & s3>0**  TR2 is a test requirement that makes a predicate **P1** in the condensation graph for Question 1, false, attempting to achieve predicate coverage. |
| **TR3** | **s1+s2<=s3 | s2+s2<=s1 | s1+s3<=s2**  TR3 is a test requirement that makes a predicate **P2** in the condensation graph for Question 1, true, attempting to achieve predicate coverage. It is only reachable if **P1** has been evaluated to false. |
| **TR4** | **s1+s2>s3 & s2+s2>s1 & s1+s3>s2**  TR4 is a test requirement that makes a predicate **P2** in the condensation graph for Question 1, false, attempting to achieve predicate coverage. It is only reachable if **P1** has been evaluated to false. |
| **TR5** | **s1==s2 & s2 ==s3**  TR5 is a test requirement that makes a predicate **P3** in the condensation graph for Question 1, true, attempting to achieve predicate coverage. It is only reachable if **P2** has been evaluated to false. |
| **TR6** | **s1 !=s2 | s2 !=s3**  TR6 is a test requirement that makes a predicate **P3** in the condensation graph for Question 1, false, attempting to achieve predicate coverage. It is only reachable if **P2** has been evaluated to false. |
| **TR7** | **s1== s2| s2==s3 | s1==s3**  TR7 is a test requirement that makes a predicate **P4** in the condensation graph for Question 1, true, attempting to achieve predicate coverage. It is only reachable if **P3** has been evaluated to false. |
| **TR8** | **s1 != s2 & s2!=s3 & s1!=s3**  TR8 is a test requirement that makes a predicate **P4** in the condensation graph for Question 1, false, attempting to achieve predicate coverage. It is only reachable if **P3** has been evaluated to false. |

**2.1(b) No, we do not think it is possible to change the condensation graph.**

**2.1(c)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case** | **Test Case Values** | **Setup Values** | **Teardown values** | **Expected Values** |
| **TC1** | **s1=-1, s2=1, s3=2** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **P1: True** |
| **TC2** | **s1= 1, s2=1, s3=2** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **P1: False**  **P2: True** |
| **TC3** | **s1= 2, s2=2, s3 =s2** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **P1: False**  **P2: False**  **P3: True** |
| **TC4** | **s1 = 2, s2 =3, s3 =2** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **P1: False**  **P2: False**  **P3: False**  **P4: True** |
| **TC5** | **s1 = 3, s2 = 4, s3 = 5** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **P1: False**  **P2: False**  **P3: False**  **P4: False** |

**2.2(a)**

|  |
| --- |
| **Clauses** |
| **C1: s1<=0** |
| **C2: s2<= 0** |
| **C3: s3<=0** |
| **C4: s1+s2<=s3** |
| **C5: s2+s3<=s1** |
| **C6: s1+s3<=s2** |
| **C7: s1==s2** |
| **C8: s2 ==s3** |
| **C9: s1==s3** |

|  |  |
| --- | --- |
| **TR1** | **s1<=0**  TR1 is a test requirement that makes a clause **C1** in the condensation graph for Question 1, true, attempting to achieve clause coverage. |
| **TR2** | **s1>0**  TR2 is a test requirement that makes a clause **C1** in the condensation graph for Question 1, false, attempting to achieve clause coverage. |
| **TR3** | **s2<=0**  TR3 is a test requirement that makes a clause **C2** in the condensation graph for Question 1, true, attempting to achieve clause coverage. |
| **TR4** | **s2>0**  TR4 is a test requirement that makes a clause **C2** in the condensation graph for Question 1, false, attempting to achieve clause coverage. |
| **TR5** | **s3<=0**  TR5 is a test requirement that makes a clause **C3** in the condensation graph for Question 1, true, attempting to achieve clause coverage. |
| **TR6** | **s3>0**  TR6 is a test requirement that makes a clause **C3** in the condensation graph for Question 1, false, attempting to achieve clause coverage. |
| **TR7** | **s1+s2<=s3**  TR7 is a test requirement that makes a clause **C4** in the condensation graph for Question 1, true, attempting to achieve clause coverage. It is only reachable if **C1, C2** and **C3** have been evaluated to false. |
| **TR8** | **s1+s2>s3**  TR7 is a test requirement that makes a clause **C4** in the condensation graph for Question 1, false, attempting to achieve clause coverage. It is only reachable if **C1, C2** and **C3** have been evaluated to false. |

|  |  |
| --- | --- |
| **TR9** | **s2+s3<=s1**  TR9 is a test requirement that makes a clause **C5** in the condensation graph for Question 1, true, attempting to achieve clause coverage. It is only reachable if **C1, C2** and **C3** have been evaluated to false. |
| **TR10** | **s2+s3>s1**  TR10 is a test requirement that makes a clause **C5** in the condensation graph for Question 1, false, attempting to achieve clause coverage. It is only reachable if **C1, C2** and **C3** have been evaluated to false. |
| **TR11** | **s1+s3<=s2**  TR9 is a test requirement that makes a clause **C6** in the condensation graph for Question 1, true, attempting to achieve clause coverage. It is only reachable if **C1, C2** and **C3** have been evaluated to false. |
| **TR12** | **s1+s3>s2**  TR10 is a test requirement that makes a clause **C6** in the condensation graph for Question 1, false, attempting to achieve clause coverage. It is only reachable if **C1, C2** and **C3** have been evaluated to false. |
| **TR13** | **s1==s2**  TR13 is a test requirement that makes a clause **C7** in the condensation graph for Question 1, true, attempting to achieve clause coverage. It is only reachable if **C4, C5** and **C6** have been evaluated to false. |
| **TR14** | **s1 != s2**  TR14 is a test requirement that makes a clause **C7** in the condensation graph for Question 1, false, attempting to achieve clause coverage. It is only reachable if **C4, C5** and **C6** have been evaluated to false. |
| **TR15** | **s2==s3**  TR15 is a test requirement that makes a clause **C8** in the condensation graph for Question 1, true, attempting to achieve clause coverage. It is only reachable if **C4, C5** and **C6** have been evaluated to false. |
| **TR16** | **s2!=s3**  TR16 is a test requirement that makes a clause **C8** in the condensation graph for Question 1, false, attempting to achieve clause coverage. It is only reachable if **C4, C5** and **C6** have been evaluated to false. |
| **TR17** | **s1==s3**  TR17 is a test requirement that makes a clause **C9** in the condensation graph for Question 1, true, attempting to achieve clause coverage. It is only reachable if **C7** or **C8** has been evaluated to false. |
| **TR18** | **s1==s3**  TR17 is a test requirement that makes a clause **C9** in the condensation graph for Question 1, false, attempting to achieve clause coverage. It is only reachable if **C7** or **C8** has been evaluated to false. |

**2.2(b)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case** | **Test Case Values** | **Setup Values** | **Teardown values** | **Expected Values** |
| **TC1** | **s1=-1, s2=1, s3=2** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C1: True**  **C2: False**  **C3: False** |
| **TC2** | **s1= 1, s2=-1, s3=0** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C1: False**  **C2: True**  **C3: True** |
| **TC3** | **s1= 1, s2=1, s3 =3** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C4: True**  **C5: False**  **C6: False** |
| **TC4** | **s1= 1, s2=3, s3 =1** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C4: False**  **C5: False**  **C6: True** |
| **TC5** | **s1= 3, s2=1, s3 =1** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C4: False**  **C5: True**  **C6: False** |
| **TC6** | **s1 = 2, s2 =2, s3 =2** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C7: True**  **C8: True** |
| **TC7** | **s1 =2, s2 = 3, s3 = 4** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C7: False**  **C8: False**  **C9: False** |
| **TC8** | **s1 =2, s2 = 3, s3 = 2** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C7: False**  **C8: False**  **C9: True** |

**2.3(a)**

|  |
| --- |
| **Clauses** |
| **C1: s1<=0** |
| **C2: s2<= 0** |
| **C3: s3<=0** |
| **C4: s1+s2<=s3** |
| **C5: s2+s3<=s1** |
| **C6: s1+s3<=s2** |
| **C7: s1==s2** |
| **C8: s2 ==s3** |
| **C9: s1==s3** |

**Truth Table for Predicate P1: C1 or C2 or C3 (T: True, F: False)**

|  |  |  |  |
| --- | --- | --- | --- |
| C1 (**s1<=0**) | C2 (**s2<= 0**) | C3 (**s3<=0**) | Outcome |
| F | F | F | \*F |
| F | F | T | \*T |
| F | T | F | \*T |
| F | T | T | T |
| T | F | F | \*T |
| T | F | T | T |
| T | T | F | T |
| T | T | T | T |

Starred rows denote the restricted active clause coverage for predicate P1. Therefore, there are four test requirements that can be seen on the table.

|  |  |
| --- | --- |
| **TR1** | **C1 & C2 & C3 = False**  TR1 is a test requirement that makes the predicate P1 in the condensation graph for Question 1, false, attempting to achieve restricted active clause coverage. |
| **TR2** | **C1 = True while C2 & C3 = False**  TR2 is a test requirement that makes the predicate P1 in the condensation graph for Question 1, true, attempting to achieve restricted active clause coverage. |
| **TR3** | **C2 = True while C1 & C3 = False**  TR3 is a test requirement that makes the predicate P1 in the condensation graph for Question 1, true, attempting to achieve restricted active clause coverage. |
| **TR4** | **C3 = True while C1 & C2 = False**  TR4 is a test requirement that makes the predicate P1 in the condensation graph for Question 1, true, attempting to achieve restricted active clause coverage. |

**Truth Table for Predicate P2: C4 or C5 or C6 (T: True, F: False, X: Impossible)**

|  |  |  |  |
| --- | --- | --- | --- |
| C4 (**s1+s2<=s3**) | C5 (**s2+s3<=s1**) | C6 (**s1+s3<=s2**) | Outcome |
| F | F | F | \*F |
| F | F | T | \*T |
| F | T | F | \*T |
| F | T | T | X |
| T | F | F | \*T |
| T | F | T | X |
| T | T | F | X |
| T | T | T | X |

Starred rows denote the restricted active clause coverage for predicate P2. Therefore, there are four additional test requirements that can be seen on the table.

|  |  |
| --- | --- |
| **TR5** | **C4 & C5 & C6 = False**  TR5 is a test requirement that makes the predicate P2 in the condensation graph for Question 1, false, attempting to achieve restricted active clause coverage. It is only reachable if **C1, C2** and **C3** have been evaluated to false. |
| **TR6** | **C4 = True while C5 & C6 = False**  TR6 is a test requirement that makes the predicate P2 in the condensation graph for Question 1, true, attempting to achieve restricted active clause coverage. It is only reachable if **C1, C2** and **C3** have been evaluated to false. |
| **TR7** | **C5 = True while C4 & C6 = False**  TR7 is a test requirement that makes the predicate P2 in the condensation graph for Question 1, true, attempting to achieve restricted active clause coverage. It is only reachable if **C1, C2** and **C3** have been evaluated to false. |
| **TR8** | **C6 = True while C4 & C5 = False**  TR8 is a test requirement that makes the predicate P2 in the condensation graph for Question 1, true, attempting to achieve restricted active clause coverage. It is only reachable if **C1, C2** and **C3** have been evaluated to false. |

**Truth Table for Predicate P3: C7 and C8 (T: True, F: False)**

|  |  |  |
| --- | --- | --- |
| C7 (**s1==s2**) | C8 (**s2 ==s3)** | Outcome |
| F | F | F |
| F | T | \*F |
| T | F | \*F |
| T | T | \*T |

Starred rows denote the restricted active clause coverage for predicate P3. Therefore, there are three additional test requirements that can be seen on the table.

|  |  |
| --- | --- |
| **TR9** | **C7 & C8 = True**  TR9 is a test requirement that makes the predicate P3 in the condensation graph for Question 1, true, attempting to achieve restricted active clause coverage. It is only reachable if **C4, C5** and **C6** have been evaluated to false. |
| **TR10** | **C7 = True & C8 = False**  TR10 is a test requirement that makes the predicate P3 in the condensation graph for Question 1, false, attempting to achieve restricted active clause coverage. It is only reachable if **C4, C5** and **C6** have been evaluated to false. |
| **TR11** | **C7 = False & C8 = True**  TR11 is a test requirement that makes the predicate P3 in the condensation graph for Question 1, false, attempting to achieve restricted active clause coverage. It is only reachable if **C4, C5** and **C6** have been evaluated to false. |

**Truth Table for Predicate P4: C7 or C8 or C9 (T: True, F: False)**

|  |  |  |  |
| --- | --- | --- | --- |
| C7 (**s1==s2**) | C8 (**s2 ==s3)** | C9 (**s1==s3)** | Outcome |
| F | F | F | \*F |
| F | F | T | \*T |
| F | T | F | \*T |
| F | T | T | T |
| T | F | F | \*T |
| T | F | T | T |
| T | T | F | T |
| T | T | T | T |

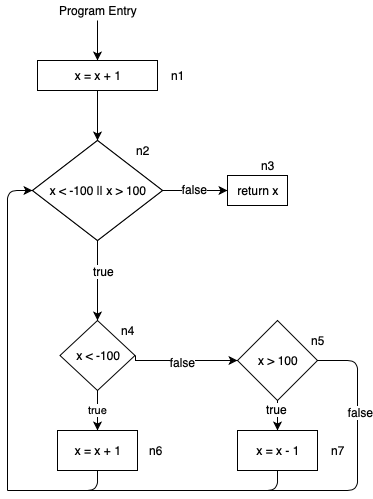
Starred rows denote the restricted active clause coverage for predicate P4. Therefore, there are four additional test requirements that can be seen on the table.

|  |  |
| --- | --- |
| **TR12** | **C7 & C8 & C9 = False**  TR12 is a test requirement that makes the predicate P4 in the condensation graph for Question 1, false, attempting to achieve restricted active clause coverage. It is only reachable if **C7** or **C8** has been evaluated to false. |
| **TR13** | **C7 = True while C8 & C9 = False**  TR13 is a test requirement that makes the predicate P4 in the condensation graph for Question 1, true, attempting to achieve restricted active clause coverage. It is only reachable if **C7** or **C8** has been evaluated to false. |
| **TR14** | **C8 = True while C7 & C9 = False**  TR14 is a test requirement that makes the predicate P4 in the condensation graph for Question 1, true, attempting to achieve restricted active clause coverage. It is only reachable if **C7** or **C8** has been evaluated to false. |
| **TR15** | **C9 = True while C7 & C8 = False**  TR15 is a test requirement that makes the predicate P4 in the condensation graph for Question 1, true, attempting to achieve restricted active clause coverage. It is only reachable if **C7** or **C8** has been evaluated to false. |

**2.3(b)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case** | **Test Case Values** | **Setup Values** | **Teardown values** | **Expected Values** |
| **TC1** | **s1=10, s2=20, s3=30** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C1: False**  **C2: False**  **C3: False** |
| **TC2** | **s1= -10, s2=20, s3=30** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C1: True**  **C2: False**  **C3: False** |
| **TC3** | **s1= 10, s2=-1,**  **s3 = 30** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C1: False**  **C2: True**  **C3: False** |
| **TC4** | **s1= 10, s2=30, s3 = 0** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C1: False**  **C2: False**  **C3: True** |
| **TC5** | **s1= 3, s2 = 4, s3 = 5** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C4: False**  **C5: False**  **C6: False** |
| **TC6** | **s1 = 3, s2 = 4, s3 = 7** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C4: True**  **C5: False**  **C6: False** |
| **TC7** | **s1 = 10, s2 = 5,**  **s3 = 5** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C4: False**  **C5: True**  **C6: False** |
| **TC8** | **s1 = 3, s2 = 7, s3 = 4** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C4: False**  **C5: False**  **C6: True** |
| **TC9** | **s1 = 1, s2 = 1, s3 = 1** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C7: True**  **C8: True** |
| **TC10** | **s1 =3, s2 = 3, s3 = 4** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C7: True**  **C8: False** |
| **TC11** | **s1 = 2, s2 = 3, s3 =3** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C7: False**  **C8: True** |
| **TC12** | **s1 =2, s2 = 3, s3 = 4** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C7: False**  **C8: False**  **C9: False** |
| **TC13** | **s1 =5, s2 = 5, s3 = 6** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C7: True**  **C8: False**  **C9: False** |
| **TC14** | **s1 =6, s2 = 5, s3 = 5** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C7: False**  **C8: True**  **C9: False** |
| **TC15** | **s1 =5, s2 = 6, s3 = 5** | **s1, s2, s3 =null** | **s1, s2, s3 =null** | **C7: False**  **C8: False**  **C9: True** |

**3(a)**



Condensation Graph for Question 3(a).

**3(b) 100% Node Coverage**

|  |  |
| --- | --- |
| **TR1** | **N1-N2-N4-N5-N7-N2-N3**  TR1 is a test requirement that visits nodes in the condensation graph for question 3, attempting to achieve node coverage. |
| **TR2** | **N1-N2-N4-N6-N2-N3**  TR2 is a test requirement that visits nodes in the condensation graph for question 3, attempting to achieve node coverage. |

Two test requirements are enough to cover 100% node coverage. This is because the first test requirement visits nodes N1-N2-N4-N5-N7-N2-N3. If N3 can be reached and passed without fail, then it can be concluded that TR1 alone covers all nodes except N6. Furthermore, to achieve 100% node coverage, second test requirement visits nodes N1-N2-N4-N6-N2-N3. If TR2 passes without a fail, TR1 and TR2 can satisfy 100% node coverage together. Consequently, test requirement set is minimal with two test requirements.

**3(c)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case** | **Test Case Values** | **Setup Values** | **Teardown values** | **Expected Values** |
| **TC1** | **X = 120** | **X = null** | **X = null** | **X = 100** |
| **TC2** | **X = -120** | **X = null** | **X = null** | **X = -100** |

**3(d)**

At their core, because of the structure of the code, predicate coverage and node coverage are essentially the same. It is not possible to reach nodes N6 and N7 in the same path during the loop. Their predicates deny them to be visited in the same iteration. If predicate coverage was applied, then it can be observed that the nodes visited in TR1 and TR2 in node coverage are visited again in the predicate coverage. Therefore, switching to predicate coverage may not yield better results.

**4. Self-Assessment**

1. Do you have a test case that represents a valid scalene triangle?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 1 | 1 | 1 | 1 | 1 |

2. Do you have a test case that represents a valid equilateral triangle?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 1 | 1 | 1 | 1 | 1 |

3. Do you have a test case that represents a valid isosceles triangle?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 1 | 1 | 1 | 1 | 1 |

4. Do you have at least three test cases that represent valid isosceles triangles such that you have tried all three permutations of two equal sides?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 0 | 0 | 0 | 0 | 1 |

5. Do you have a test case in which one side has a zero value?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 1 | 1 | 0 | 1 | 1 |

6. Do you have a test case in which one side has a negative value?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 0 | 0 | 1 | 1 | 1 |

7. Do you have a test case with three integers such that the sum of two is equal to the third?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 0 | 0 | 1 | 1 | 1 |

8. Do you have at least three test cases in category 7 such that you have tried all three permutations where the length of one side is equal to the sum of the lengths of the other two sides?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 0 | 0 | 0 | 0 | 1 |

9. Do you have a test case with three integers greater than zero such that the sum of two numbers is less than the third?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 1 | 1 | 1 | 1 | 1 |

10. Do you have at least three test cases in category 9 such that you have tried all three permutations?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 0 | 0 | 0 | 1 | 0 |

11. Do you have a test case in which all sides are zero?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 0 | 0 | 0 | 0 | 0 |

12. Do you have at least one test case specifying non-integer values? \*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 0 | 0 | 0 | 0 | 0 |

13. Do you have at least one test case specifying the wrong number of values (2 or less, four or more) \*\*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 0 | 0 | 0 | 0 | 0 |

14. For each test case, did you specify the expected output from the program in addition to the input values?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 1 | 1 | 1 | 1 | 1 |

**Total Results:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NC | EC | PC | CC | RACC |
| 6 | 6 | 7 | 9 | 10 |

**Which coverage model achieves the highest score?**

RACC followed by CC. RACC in nature tends to achieve a higher grade than others because it tends to cover permutations of variables in nature than others.