

Лабораториска 3 – Logic coverage 211055

```
public boolean isTriangle(int side1, int side2, int side3) {
    return side1+side2>side3 && side1+side3>side2 && side2+side3>side1;
}
```

1) Да се напишат предикатите кои произлегуваат од кодот, како и клаузулите кои што се дел од тие предикати.

- Предикат кој произлегува од овој код е : $side1+side2>side3 \ \&\& \ side1+side3>side2 \ \&\& \ side2+side3>side1$;
- Постојат 3 клаузули во овој предикат:
 - $side1+side2>side3$;
 - $side1+side3>side2$;
 - $side2+side3>side1$;

2) Да се испишат соодветните табели на вистинитост за предикатите.

	(side1+side2>side3) && (side1+side3>side2) && (side2+side3>side1)						
	a	b	c	P			
	side1+side2>side3	side1+side3>side2	side2+side3>side1	(side1+side2>side3) && (side1+side3>side2) && (side2+side3>side1)	Pa	Pb	Pc
1	T	T	T	T	T	T	T
2	T	T	F	F			T
3	T	F	T	F		T	
4	T	F	F	F			
5	F	T	T	F	T		
6	F	T	F	F			
7	F	F	T	F			
8	F	F	F	F			

Truth Table:

Row#	a	b	c	P	Pa	Pb	Pc
1	T	T	T	T	T	T	T
2	T	T	F				T
3	T		T			T	
4	T						
5		T	T		T		
6		T					
7			T				
8							

3) Да се најдат TR за следните покривања: GACC, RACC

- GACC:
 - За GACC да биде покриено потребно е мајор клаузулите да бидат различни односно еднаш T еднаш F, а минорните клаузули не мора да бидат исти. Во нашиот случај ги имаме паровите:

$a \rightarrow (1,5)$
 $b \rightarrow (1,3)$
 $c \rightarrow (1,2)$

The following result for GACC is based on the truth table on the right:

Major Clause	Set of possible tests
a	(1,5)
b	(1,3)
c	(1,2)

➤ RACC:

- За RACC да биде покриено потребно е мајор клаузулите да бидат различни односно еднаш Т еднаш F, а минорните клаузули мора да бидат исти. Во нашиот случај ги имаме паровите:
 $a \rightarrow (1,5)$
 $b \rightarrow (1,3)$
 $c \rightarrow (1,2)$

The following result for RACC is based on the truth table on the right:

Major Clause	Set of possible tests
a	(1,5)
b	(1,3)
c	(1,2)

4) Да се најдат тест вредности (генерирајте параметризирани тестови со JUnit) кои што задоволуваат RACC.

Валидни тестови 1, 2, 3, 5

```

Triangle.java x TriangleRACC.java
2 usages
1 public class Triangle {
2
3     4 usages
4     public boolean isTriangle(int side1, int side2, int side3){
5         return side1+side2>side3 && side1+side3>side2 && side2+side3>side1;
6     }
7 }

```

```
Triangle.java  TriangleRACC.java x
1  import org.junit.*;
2  import static org.junit.Assert.*;
3  import java.util.*;
4
5  public class TriangleRACC {
6
7      //tests, 1(T T T: T), 2(T T F: F), 3(T F T: F), 5(F T T: F)
      5 usages
8      Triangle triangle;
9      @Before
10     public void setUp(){
11         triangle = new Triangle();
12     }
13
14     @Test
15     public void test1() throws Exception{
16         assertTrue(triangle.isTriangle(3,4,5)); //1(T T T)
17     }
18     @Test
19     public void test2() throws Exception{
20         assertFalse(triangle.isTriangle(10,2,3)); //2(T T F)
21     }
22     @Test
23     public void test3() throws Exception{
24         assertFalse(triangle.isTriangle(5,7,1)); //3(T F T)
25     }
26     @Test
27     public void test5() throws Exception{
28         assertFalse(triangle.isTriangle(2,4,8)); //5(F T T)
29     }
30 }
31
```