

СОФТВЕРСКИ КВАЛИТЕТ И ТЕСТИРАЊЕ

ДОМАШНА ЗАДАЧА 7

Фисник Лимани, 151027

Го имаме кодот:

```
public static int patternIndex (String subject, String pattern)
{
    final int NOTFOUND = -1;
    int iSub = 0, rtnIndex = NOTFOUND;
    boolean isPat = false;
    int subjectLen = subject.length();
    int patternLen = pattern.length();

    while (isPat == false && iSub + patternLen - 1 < subjectLen)
    {
        if (subject.charAt(iSub) == pattern.charAt(0))
        {
            rtnIndex = iSub; // Starting at zero
            isPat = true;
            for (int iPat = 1; iPat < patternLen; iPat++)
            {
                if (subject.charAt(iSub + iPat) != pattern.charAt(iPat))
                {
                    rtnIndex = NOTFOUND;
                    isPat = false;
                    /* MB: isPat = true; */
                    break; // out of for loop
                }
            }
            iSub++;
        }
        return (rtnIndex);
    }
}
```

а) Да се напишат предикатите кои произлегуваат од следниот код

1. `isPat == false && iSub + patternLen - 1 < subjectLen`
2. `subject.charAt(iSub) == pattern.charAt(0)`
3. `iPat < patternLen`
4. `subject.charAt(iSub + iPat) != pattern.charAt(iPat)`

b) Да се испишат соодветните таблици на вистинитост за предикатите

Функцијата има два влезни параметри:

- String subject
- String pattern

И ги има овие внатрешни променливи:

- final int NOTFOUND = -1
- int iSub = 0
- int rtnIndex = NOTFOUND
- boolean isPat = false
- int subjectLen = subject.length();
- int patternLen = pattern.length();

Прво го испитаме reachability:

- Првиот предикат:
 - o е секогаш достапен (reachable)
- Вториот предикат:
 - o Ако првиот предикат е true
 - o т.е. **isPat == false && iSub + patternLen - 1 < subjectLen**
- Третиот предикат:
 - o Ако првиот и вториот предикат се true
 - o т.е. **isPat == false && iSub + patternLen - 1 < subjectLen**
&&
subject.charAt(iSub) == pattern.charAt(0)
- Четвртиот предикат:
 - o Ако првиот, вториот и третиот предикат се true
 - o т.е. **isPat == false && iSub + patternLen - 1 < subjectLen**
&&
subject.charAt(iSub) == pattern.charAt(0)
&&
iPat < patternLen

1. Табела на вистинитост за предикатот:

isPat == false && iSub + patternLen - 1 < subjectLen

Нека го направиме замената:

- isPat == false => a
- iSub + patternLen - 1 < subjectLen => b

	a	b	p_a	p_b
1	T	T	T	T
2	T	F		T
3	F	T	T	
4	F	F		

2. Табела на вистинитост за предикатот:

subject.charAt(iSub) == pattern.charAt(0)

Нека го направиме замената:

- subject.charAt(iSub) == pattern.charAt(0) => c

	c	p_c
1	T	T
2	F	F

3. Табела на вистинитост за предикатот:

iPat < patternLen

Не го направиме замената:

- iPat < patternLen => d

	d	p_d
1	T	T
2	F	F

4. Табела на вистинитост за предикатот:

subject.charAt(iSub + iPat) != pattern.charAt(iPat)

Нека го направиме замената:

- subject.charAt(iSub + iPat) != pattern.charAt(iPat) => e

	e	p_e
1	T	T
2	F	F

с) Да се најдат TR за следните покривања: CACC, RACC

TR за првиот предикат за CACC и RACC: $(a,b)=\{(T,T), (T,F), (F,T)\}$

TR за вториот предикат за CACC и RACC: $c = \{T, F\}$

TR за третиот предикат за CACC и RACC: $d = \{T, F\}$

TR за третиот предикат за CACC и RACC: $e = \{T, F\}$

CORRELATED & RESTRICTIVE ACTIVE CLAUSE COVERAGE

Predicate	CLAUSES		INPUTS		OUTPUT
			subject	pattern	
a&&b	T	T	a	a	0
	F	t	Infeasible (isPat is false)		
	t	F	a	ab	-1
c	T	-	a	a	0
	F	-	a	c	-1
d	T	-	ab	ac	-1
	F	-	a	a	0
e	T	-	ab	ac	-1
	F	-	ab	ab	0

d) Да се најдат тест вредности (генерирајте параметризирани тестови со JUnit) кои што задоволуваат RACC. Внимавајте да биде задоволено reachability.

```
@ParameterizedTest
@MethodSource("values")
void testPatternIndex(String subject, String pattern, int expectedResult) {
    System.out.println("Parameterized parameters are: (" + subject + ", " + pattern + ")");
    assertEquals(expectedResult, Domashna7.patternIndex(subject, pattern));
}

public static Collection<Object[]> values(){
    return Arrays.asList(new Object[][]{
        {"a", "a", 0},
        {"a", "ab", -1},
        {"a", "c", -1},
        {"ab", "ac", -1},
        {"ab", "ab", 0},
    });
}
```

Results:

- ▼ ✓ Test Results
 - ▼ ✓ Domashna7Test
 - ▼ ✓ testPatternIndex(String, String, int)
 - ✓ [1] a, a, 0
 - ✓ [2] a, ab, -1
 - ✓ [3] a, c, -1
 - ✓ [4] ab, ac, -1
 - ✓ [5] ab, ab, 0