Nismo ispočetka znali da slideovi neće biti objavljeni pa smo krenuli slikat od 6. Zadatka. Na prvoj slici su rješenja.

Odgovor na 6. Je B

PRIPREMA ZA Z1

(1.) nemamo konstruktor bez parametara. Ne moramo ga imati ako ne bi imali ovaj s parametrima, tad se on implicitno dodaje

(2.) ne može se importati cijeli paket

(3.) D) zato jer za neprovjeravanu ne treba try-catch
A - ne javlja jer je neprovjeravana (za nju ne try-catch)
B - nema smisla, C - ne

(4.) A)

(5.) 1.+, 2.+, 3. modeliranje, 4. X predikate

(7.) A) ne dohvaćamo proizvođača
B) ne selektiramo sve (/*)xne
C) ✓
D) isto kao B

(9.) A) ✓

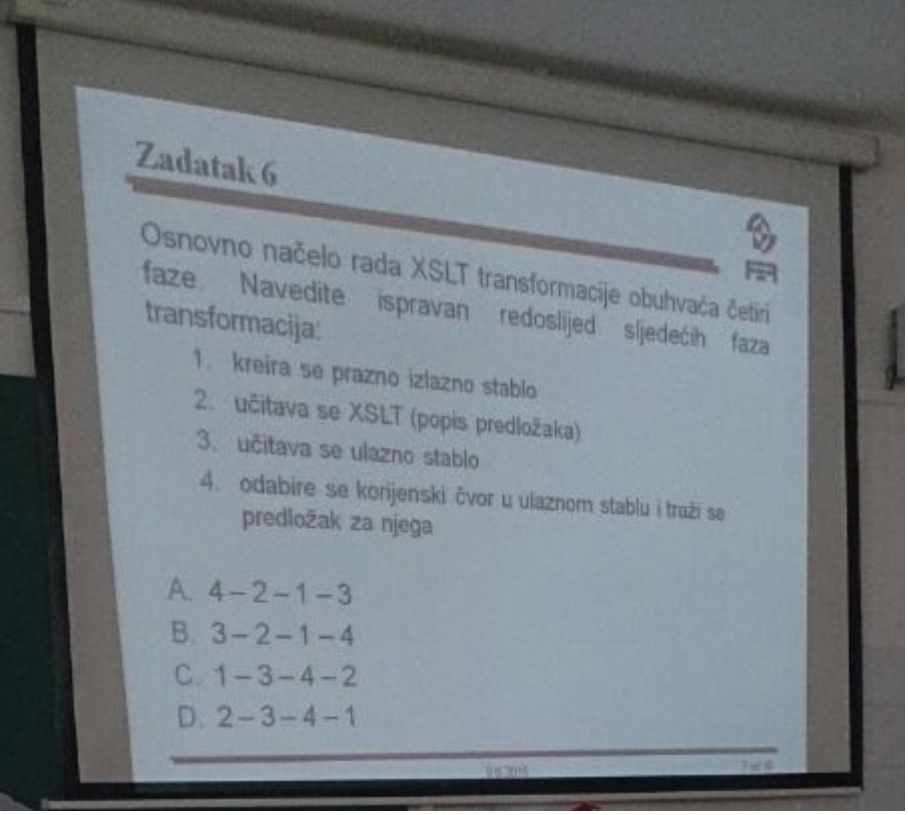
(10.) C) ✓

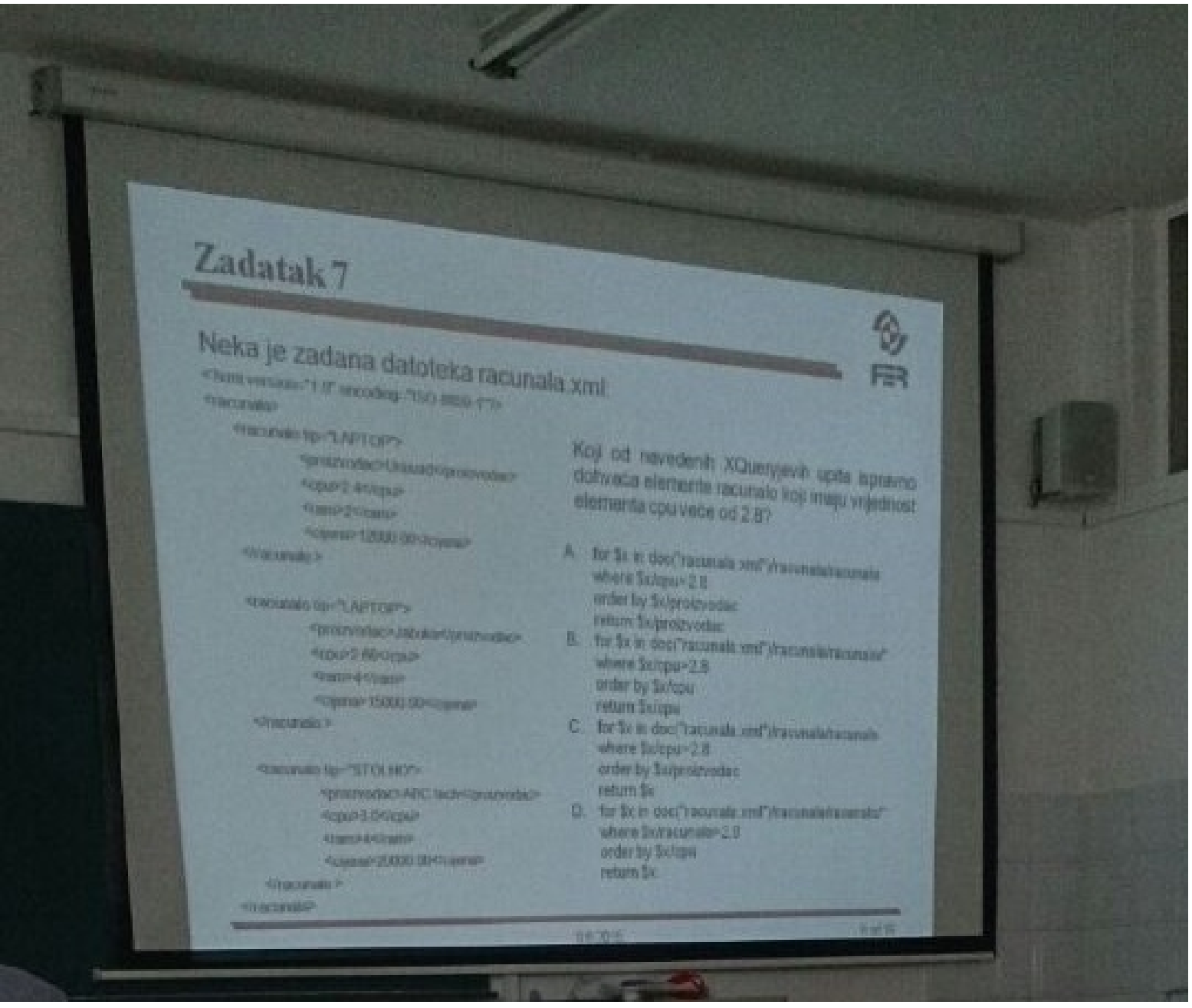
(11.) D) ✓ - try-catch ne hvata errore koje baca assert

(12.) B) br. lukova - br. čvorova - 2

(14.) B) labela 1 !zahtjev, labela 2 ?zahtjev
$a_0 b_0 - a_0 b_1$

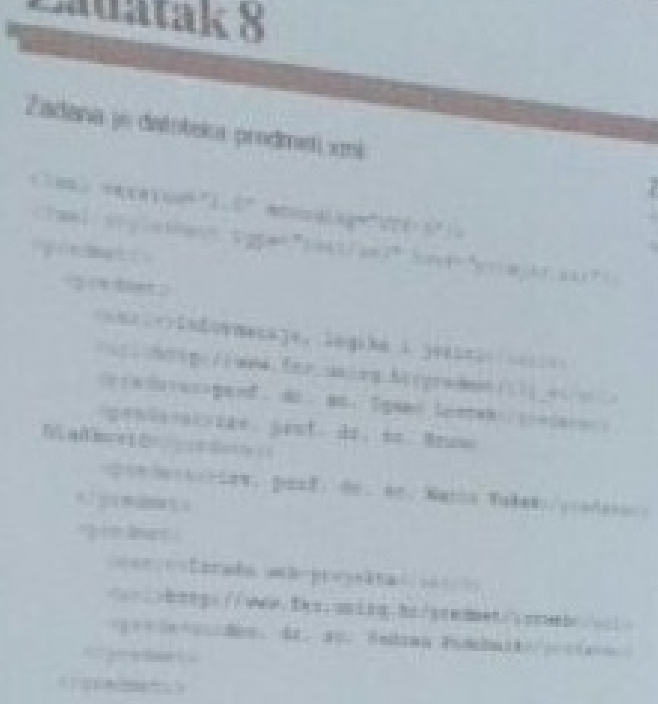
KOMBINATORNO TESTIRANJE! Z1


Zadatak 6
Osnovno načelo rada XSLT transformacije obuhvaća četiri faze. Navedite ispravan redoslijed sljedećih faza transformacija:
1. kreira se prazno izlazno stablo
2. učitava se XSLT (popis predložaka)
3. učitava se ulazno stablo
4. odabire se korijenski čvor u ulaznom stablu i traži se predložak za njega
A. 4 – 2 – 1 – 3
B. 3 – 2 – 1 – 4
C. 1 – 3 – 4 – 2
D. 2 – 3 – 4 – 1

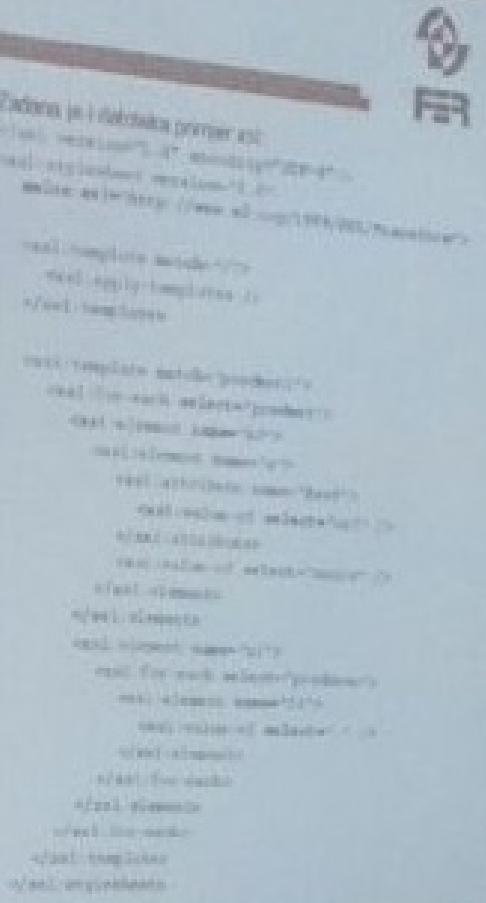

Zadatak 7
Neka je zadana datoteka racunala.xml
Koji od navedenih XQueryjevih upita ispravno dohvaća elemente racunalo koji imaju vrijednost elementa cpu veće od 2.8?
A. for $x in doc("racunala.xml")/racunala/racunalo
where $x/cpu > 2.8
order by $x/proizvodac
return $x/proizvodac
B. for $x in doc("racunala.xml")/racunala/racunalo
where $x/cpu>2.8
order by $x/cpu
return $x/cpu
C. for $x in doc("racunala.xml")/racunala/racunalo
where $x/cpu>2.8
order by $x/proizvodac
return $x
D. for $x in doc("racunala.xml")/racunala/racunalo
where $x/racunalo>2.8
order by $x/cpu
return $x

Zadatak 8

Zadana je datoteka predmeti.xml:



Zadana je i datoteka primjer.xsl:



Kako izgleda izlazna datoteka, nastala primjenom XSL-a primjer.xsl na podatke u datoteci predmeti.xml?

Zadatak 8

```
<?xml version="1.0" encoding="UTF-8"?>
<h3>
  <a href="http://www.fer.unizg.hr/predmet/ilj_a">Informacija, logika i jezici</a>
</h3>

<ul>
  <li>prof. dr. sc. Ignac Lovrek</li>
  <li>izv. prof. dr. sc. Bruno Blašković</li>
  <li>izv. prof. dr. sc. Mario Kušek</li>
</ul>

<h3>
  <a href="http://www.fer.unizg.hr/predmet/izrweb">Izrada web-projekta</a>
</h3>

<ul>
  <li>doc. dr. sc. Vedran Podobnik</li>
</ul>
```

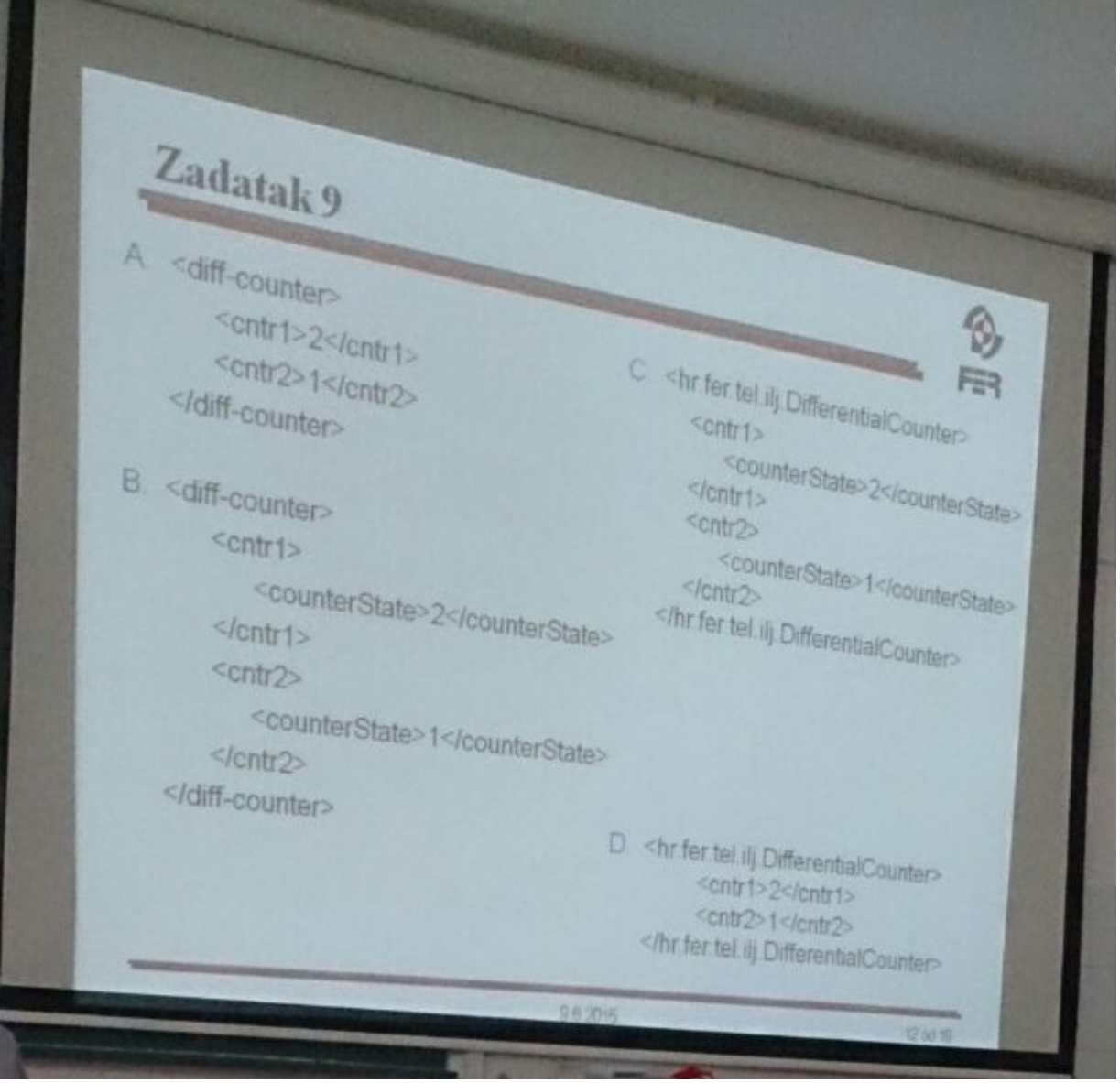
Zadatak 9

Što će se ispisati izvršavanjem sljedećeg programskog odsječka?

```
public static void main(String[] args ) {
    DifferentialCounter dc1 = new DifferentialCounter();
    XStream xs = new XStream();
    xs.alias("diff-counter", DifferentialCounter.class);
    String xml = xs.toXML(dc1);
    System.out.println(xml);
    DifferentialCounter dc2 = (DifferentialCounter) xs.fromXML(xml);
}
```

Klasa DifferentialCounter zadana je ovako:

```
package hr.fer.tel.ilj;
public class DifferentialCounter {
    private int cntr1;
    private int cntr2;
    public DifferentialCounter () {
        this.cntr2 = 1;
        this.cntr1 = 2;
    }
    ...
}
```


Zadatak 9
A
<diff-counter>
<cntr1>2</cntr1>
<cntr2>1</cntr2>
</diff-counter>
B
<diff-counter>
<cntr1>
<counterState>2</counterState>
</cntr1>
<cntr2>
<counterState>1</counterState>
</cntr2>
</diff-counter>
C
<hr.fer.tel.ilj.DifferentialCounter>
<cntr1>
<counterState>2</counterState>
</cntr1>
<cntr2>
<counterState>1</counterState>
</cntr2>
</hr.fer.tel.ilj.DifferentialCounter>
D
<hr.fer.tel.ilj.DifferentialCounter>
<cntr1>2</cntr1>
<cntr2>1</cntr2>
</hr.fer.tel.ilj.DifferentialCounter>

Zadatak 10

Odredite točan redoslijed dijelova životnog ciklusa EasyMockovog lažnog objekta:

1. CREATE
2. VERIFY
3. REPLAY
4. EXPECT

A. 1, 2, 3, 4
B. 1, 3, 4, 2
C. 1, 4, 3, 2
D. Redoslijed se ne može definirati.

Zadatak 11

Što od navedenog opisuje metodu

```
assertEquals (java.lang.String message,
java.lang.Object expected, java.lang.Object actual)
```

A. Vraća podatak tipa true ili false.

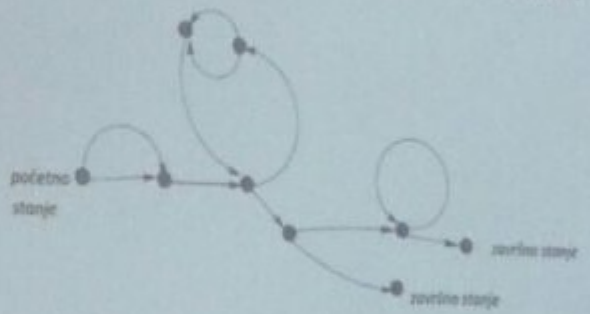
B. Uspoređuje navedena tri argumenta metode.

C. Vraća rezultat usporedbe objekata expected i actual.

D. Ne vraća ništa (void).

Zadatak 12

Koliko je minimalno putova potrebno testirati kako bi se ispitala svaka grana i svaki čvor programa opisanog sljedećim grafom stanja s označenim početnim i završnim stanjima programa?

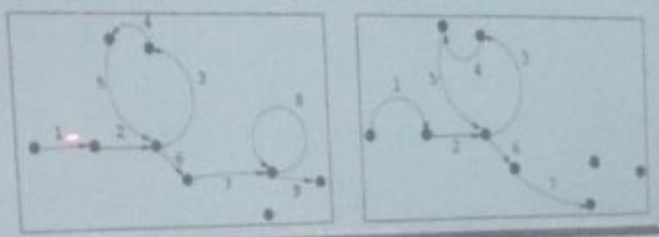


A. 1

B. 2

C. 9

D. 12



Zadatak 13

Što opisuje sljedeći zapis u Floyd-Hoare logici?

```
P={p|m_dnsQuestion("www.ietf.org")}|
C=...metode za dohvat IP adrese...
Q={p|m_dnsAnswer("4.31.198.44")}|
```

ako su oba predikata istinita, specificira što rade metode za dohvat IP adrese

ako nisu istiniti, opisuje ponašanje metoda za neregularne slučajeve (ne smije doći do zastoja!)

Što opisuje sljedeći zapis u Floyd-Hoare logici?

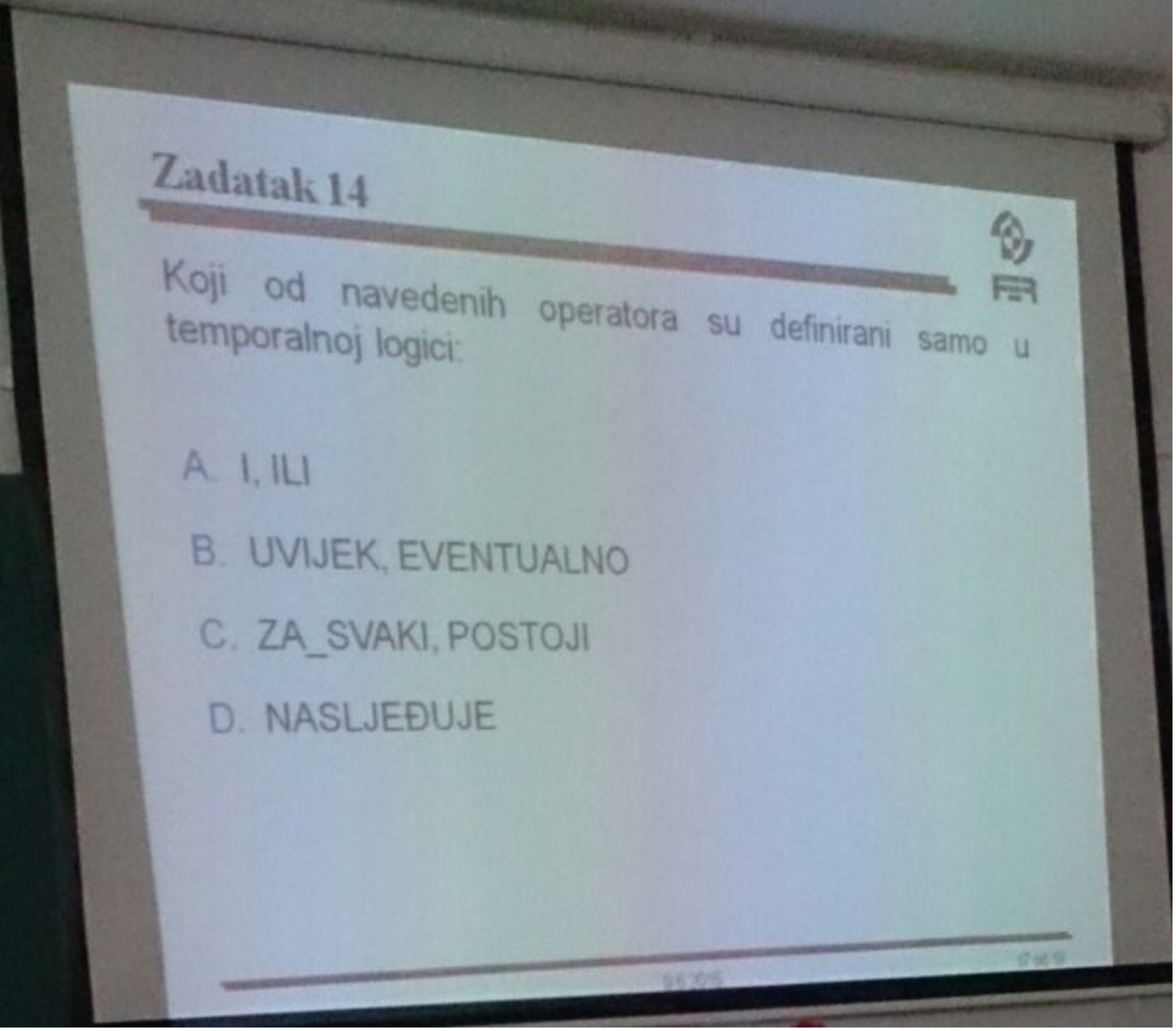
P={p|m_dnsQuestion("www.ietf.org")}|
C=...metode za dohvat IP adrese...
Q={p|m_dnsAnswer("4.31.198.44")}|

ako su oba predikata istinita, specificira što rade metode za dohvat IP adrese

ako nisu istiniti, opisuje ponašanje metoda za neregularne slučajeve (ne smije doći do zastoja!)

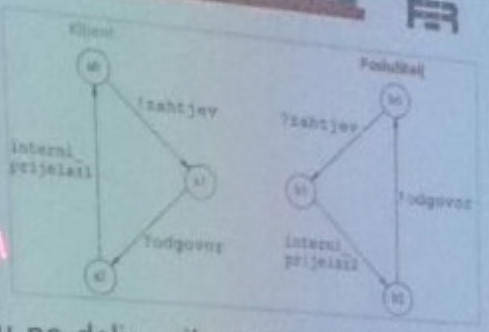
Formalno:

Ako je tvrdnja *P* točna prije izvršenja naredbe *C* onda je tvrdnja *Q* točna ili je program u blokadi!

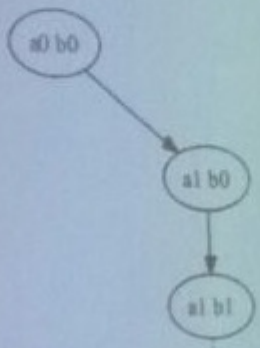

Zadatak 14
Koji od navedenih operatora su definirani samo u temporalnoj logici:
A. I, ILI
B. UVIJEK, EVENTUALNO
C. ZA_SVAKI, POSTOJI
D. NASLJEĐUJE

Zadatak 15

Na slici je prikazan *klijent-poslužitelj* sustav, zadan kao 2 procesa koji komuniciraju razmjenom poruka (operator ! označava slanje poruke, a operator ? primanje poruke).



Nadopisati labele koje nedostaju na dolje prikazanom *dijelu* grafa interakcije sustava!



Zadatak 16

Zadana je 3-SAT formula u konjuktivnoj normalnoj formi (CNF):

$(x_{11}$ OR x_{12} OR $x_{13})$ AND
$(x_{21}$ OR x_{22} OR $x_{23})$ AND
(1 OR 0 OR 0)

Pridjeliti vrijednosti varijabli $x_{11}, \ldots, x_{23}$ tako da sustav bude zadovoljiv (*SAT - satisfiable*)!

Zadana je 3-SAT formula u konjuktivnoj normalnoj formi (CNF):

$(x_{11}$ OR x_{12} OR $x_{13})$ AND
$(x_{21}$ OR x_{22} OR $x_{23})$ AND
(1 OR 0 OR 0)

Pridjeliti vrijednosti varijabli $x_{11}, \ldots, x_{23}$ tako da sustav bude zadovoljiv (*SAT - satisfiable*)!

primjer rješenja:

$x_{11}=1$, $x_{12}=0$, $x_{13}=0$, $x_{21}=1$, $x_{22}=0$, $x_{23}=0$

provjera:

(1 OR 0 OR 0) AND (1 OR 0 OR 0) AND (1 OR 0 OR 0) = 1 ✓