

Informacija, logika i jezici

Drugi međuispit 2009./10.

- 1) Koji Java razredi omogućavaju pristup „sirovim“ tokovima podataka?
 - ✓ `InputStream` i `OutputStream`.
- 2) Što je XML?
 - ✓ XML je proširljivi jezik za označavanje.
- 3) Kako izgleda XML deklaracija?
 - ✓ `<?xml version="1.0" encoding="UTF-8"?>`
- 4) Kada XML deklaracija mora sadržavati `standalone="yes"`?
 - ✓ Kada XML koristi vanjski dokument za strukturiranje (DTD).
- 5) Što je DOM?
 - ✓ DOM je model koji predstavlja sadržaj XML-a u obliku stabla.
- 6) Što je SAX i kako radi?
 - ✓ SAX je programsko sučelje koje tijekom čitanja XML dokumenta generira događaje i prosljeđuje ih objektu za obradu.
- 7) Koje od sljedećih tvrdnji su istinite za valjanje XML dokumente?
 - Dokument je valjan ako zadovoljava osnovna pravila XML-a.
 - Dokument je valjan ako zadovoljava dodatna pravila specifičnog XML jezika.
 - Dodatna pravila specifičnog XML jezika provjeravaju se nakon parsiranja.
 - ✓ **Sve tvrdnje su točne.**
- 8) Čemu služi metoda `alias` razreda `XStream`?
 - ✓ Za promjenu imena oznake iz `paket`. Razred u prikladniji oblik.
- 9) Koji od sljedećih XML elemenata je dobro oblikovan?
 - `<uml_diagram "type"="static">Class</uml_diagram>`
 - `<uml_diagram type=static>Class</uml_diagram>`
 - `<uml_diagram type=static name=class/>`
 - ✓ **`<uml_diagram type="static">Class</uml_diagram>`**

10) Koji od sljedećih XML dokumenata je dobro oblikovan?

- `<first>prvi</first> <second>drugi</drugi>`
- `<first>prvi <second>drugi</first> </second>`
- ✓ **`<first>prvi <second>drugi</second> </first>`**

11) Čemu služi XML Namespace?

- ✓ Omogućava svakom XML jeziku definiranje vlastitog prostora imena.

12) Koje je vrste element definiran XML shemom `<xs:element name="ime"/>`?

- `string`
- ✓ **`anyType`**

13) Koji XPath izraz će odabrati sve `book` elemente XML dokumenta?

- ✓ `//book`

14) Koji XPath izraz će odabrati sve `title` elemente koji imaju atribut `lang`?

- ✓ `//title[@lang]`

15) Navedite redoslijed faza tijekom rada XSLT transformacije.

1. Stvara 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.

- ✓ `3 – 2 – 1 – 4`

16) Koji od sljedećih XML komentara je ispravan po W3C specifikaciji?

- `<!-- ovo je komentar -- i ovo isto -->`
- `<p <!-- size="15" --> />`
- `<!-- prvi <!-- drugi --> komentar -->`
- ✓ **`<!-- ovo je komentar <xml:comment> -->`**

17) Koji će biti rezultat izvođenja XQuery upita nad sljedećim dokumentom koji se nalazi u datoteci `arrays.xml`?

```
<arrays>

  <array nc="1">
    <element value="100"/>
    <element value="200"/>
  </array>

  <array nc="2">
    <element value="300"/>
    <element value="400"/>
  </array>

</arrays>
```

```
element { concat("array", doc("arrays.xml")/@nc) }
{
  doc("arrays.xml")/arrays/array/element[@value >= 200]
}
```

•

```
<arrays>
  <array>
    <element value="200"/>
  </array>
  <array>
    <element value="300"/>
    <element value="400"/>
  </array>
</arrays>
```

✓

```
<array>
  <element value="200"/>
  <element value="300"/>
  <element value="400"/>
</array>
```

18) XML stablo zadano je slikom. Trebalo je odabrati koji XML dokument odgovara stablu. Trebalo je paziti na sljedeće:

- Dokument mora biti dobro oblikovan.
- Čvorovi djece moraju biti navedeni u istom redoslijedu kao na slici.
- Čvor mora imati istog roditelja kao na slici.

19) Sljedeću XML shemu prevedite u DTD:

```
<xs:element name="email">
  <xs:complexType>
    <xs:sequence maxOccurs="unbounded">
      <xs:element name="head"/>
      <xs:element name="body"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

- <!ELEMENT email (head | body)+ >
- <!ELEMENT email (head+, body+) >
- ✓ <!**ELEMENT email (head, body)+** >

20) Za zadani XML dokument koji se nalazi u datoteci dokument.xml koji XQuery upit će vratiti karakteristike svih računala čiji procesor ima 2.8 Ghz ili više?

```
<racunala>
  <racunalo>
    <naziv>Prvi</naziv>
    <cpu>2.2</cpu>
    <ram>512</ram>
  </racunalo>

  <racunalo>
    <naziv>Drugi</naziv>
    <cpu>3.6</cpu>
    <ram>2048</ram>
  </racunalo>
</racunala>
```

```
• for $x in doc("dokument.xml")/racunala/racunalo/*
  where $x/cpu >= 2.8
  return $x
```

```
• for $x in doc("dokument.xml")/racunala/racunalo/
  where $x/cpu >= 2.8
  return $x/cpu
```

```
✓ for $x in doc("dokument.xml")/racunala/racunalo
  where $x/cpu >= 2.8
  return $x
```