



Parte II			
Cognome		Laureando	si <input type="checkbox"/> no <input type="checkbox"/>
Nome		Matricola	

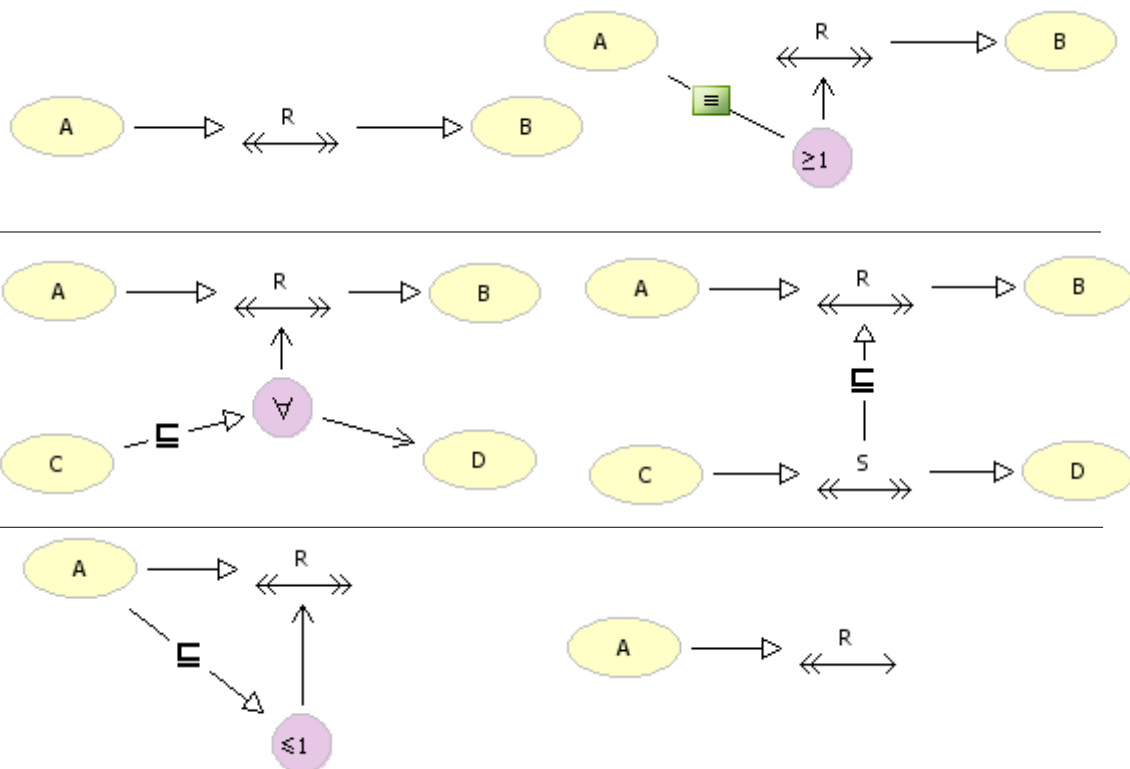
<b>4</b> 8 punti	<p>Si descrivano vantaggi e svantaggi dei due approcci all'interfaccia verso documenti XML: SAX, DOM e JAXB.</p> <p>Si proponga il DTD per un linguaggio XML per l'invio di fatture, indicando per ogni approssimazione necessaria se questa può essere evitata utilizzando XSD. Indicare anche laddove l'uso di XSD renderebbe più semplice o meglio strutturata la stesura dello schema.</p> <p>Ogni file contiene una o più fatture, ciascuna identificata da:</p> <ul style="list-style-type: none"><li>• numero del documento</li><li>• dati mittente (denominazione, indirizzo, p.iva)</li><li>• dati destinatario (denominazione, indirizzo, p.iva)</li><li>• una o più righe di dettaglio, ciascuna riportante<ul style="list-style-type: none"><li>◦ la descrizione del prodotto tramite il linguaggio <a href="http://www.demo.it/prodotti">http://www.demo.it/prodotti</a></li><li>◦ prezzo unitario</li><li>◦ quantità</li></ul></li><li>• l'aliquota iva</li></ul> <p>Si sviluppi un programma che, ricevendo in ingresso un file di grosse dimensioni tramite uno stream di rete (ignorare i dettagli della ricezione del file), stampa a video il totale ivato per ogni fattura.</p>														
<p>Punto 1) vedi slide</p> <p>Punto 2)</p> <table border="0"><tr><td>&lt;!ELEMENT archivio (fattura+)&gt;</td><td>&lt;!ELEMENT fattura (mittente, destinatario dettaglio+)&gt;</td></tr><tr><td>&lt;!ELEMENT mittente EMPTY&gt;</td><td>&lt;!ELEMENT destinatario EMPTY&gt;</td></tr><tr><td>&lt;!ELEMENT dettaglio descrizione&gt;</td><td>&lt;!ELEMENT descrizione ANY&gt;</td></tr><tr><td>&lt;!ATTLIST      fattura            numero            ID            #REQUIRED</td><td></td></tr><tr><td></td><td>iva            CDATA #REQUIRED&gt;</td></tr><tr><td>&lt;!ATTLIST      dettaglio        prezzo            CDATA #REQUIRED</td><td></td></tr><tr><td></td><td>numero            CDATA #REQUIRED&gt;</td></tr></table> <p>non posso esprimere i tipi numerici (iva, prezzo e numero), come invece potrei in XSD. Non posso indicare che la descrizione sarà espressa tramite un particolare linguaggio, in XSD utilizzerei XSD:any. Inoltre in XSD potrei definire un tipo unico per gli elementi mittente e destinatario.</p> <p>Punto 3) uso SAX per l'assunzione di file di grandi dimensioni</p> <pre>public class MyHandler extends DefaultHandler{     float aliquota, totale;     void startElement(String namespaceURI,String localName, String qName, Attributes atts){         if (localName.equals("fattura")){             totale = 0; aliquota = Float.parseFloat(atts.getValue("iva"));        }             if (localName.equals("dettaglio")){                 totale += (1+aliquota/100)*Float.parseFloat(atts.getValue("prezzo"))*                     Float.parseFloat(atts.getValue("numero"));        }         }     }     void endElement(String namespaceURI,String localName, String qName){         if (localName.equals("fattura")) System.out.println(totale);     } }</pre>		<!ELEMENT archivio (fattura+)>	<!ELEMENT fattura (mittente, destinatario dettaglio+)>	<!ELEMENT mittente EMPTY>	<!ELEMENT destinatario EMPTY>	<!ELEMENT dettaglio descrizione>	<!ELEMENT descrizione ANY>	<!ATTLIST      fattura            numero            ID            #REQUIRED			iva            CDATA #REQUIRED>	<!ATTLIST      dettaglio        prezzo            CDATA #REQUIRED			numero            CDATA #REQUIRED>
<!ELEMENT archivio (fattura+)>	<!ELEMENT fattura (mittente, destinatario dettaglio+)>														
<!ELEMENT mittente EMPTY>	<!ELEMENT destinatario EMPTY>														
<!ELEMENT dettaglio descrizione>	<!ELEMENT descrizione ANY>														
<!ATTLIST      fattura            numero            ID            #REQUIRED															
	iva            CDATA #REQUIRED>														
<!ATTLIST      dettaglio        prezzo            CDATA #REQUIRED															
	numero            CDATA #REQUIRED>														
<p style="text-align: right;">continuare sulla pagina 4</p>															

5

6 punti

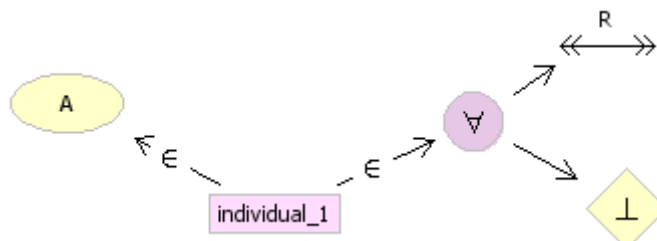
Per ciascuna coppia di modelli GrOWL:

- si indichi il potere espressivo di ciascuna frammento (RDFS / OWL-Lite / OWL-DL)
- si dica se sono equivalenti (limitatamente ai termini comuni)
- nel caso non lo fossero, si proponga un modello corretto rispetto ad uno solo dei due.



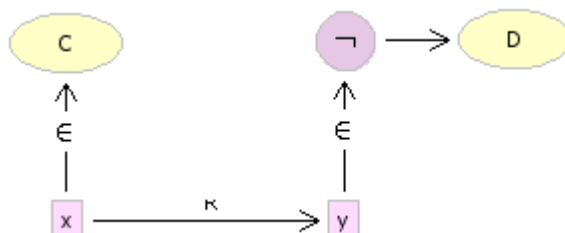
Coppia 1:

A: RDFS; B : OWL-DL non sono equivalenti (l'avere un R è necessaria e sufficiente per A nel primo, solo sufficiente nel secondo). Modello compatibile sono con il secondo frammento:



Coppia 2:

A: OWL-DL; B : RDFS non sono equivalenti. Nel secondo un elemento di C può comunque avere un R che non sta in D. Modello compatibile sono con il secondo:



Coppia 3:

A: OWL-DL; B : OWL-Lite  
sono equivalenti

## Allegati

### DOM

```
DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
DocumentBuilder builder = factory.newDocumentBuilder();
Document doc = builder.parse("libro.xml");
```

```
Interface Document{...
    Element getDocumentElement();
}
Interface Element extends Node {...
    String getAttribute(String name) ;
    public Node[ ] getChildNodes();
    String getTagName() ;
}
Interface CharacterData extends Node{...
    String getData();
}
```

### SAX

```
Interface ContentHandler {...
    void startDocument();
    void startElement(String namespaceURI,String localName, String qName, Attributes atts);
    void characters(char[] ch, int start, int length);
    void endDocument();
    void endElement(String namespaceURI,String localName, String qName);
}
Class DefaultHandler Implements ContentHandler();
interface Attributes{...
    String getValue(String qName);
}
```

