Homework #1 Solutions

1. 3pts each, 30 pts. total
   a) <a><b/><a>
      No, the last <a> should be </a> in order to close the element
   b) <a x=”1” X=”2”><b>foo</b></a>
      Note, since XML is case-sensitive, a’s attributes are distinct
   c) <a><b>foo</b></a><a>bar</a>
      No, this document has two roots
   d) <a><b>foo</b><b/>bar</a>
e) `<a x="1">`<b x="2">foo</b>`<</a>

*Note order of attributes with respect to child elements is insignificant*

```
  +---+                     +---+
  |   |                     |   |
  +---+                     +---+
      |                     |   |
      |                     | x |
      |                     | @ |
      +---+                   +---+
            +---+               +---+
            |   |               |   |
            +---+               +---+
                |               |   |
                +---+           |   |
                    foo       x 1
                +---+           |   |
                    2             +---+
```

f) `<a>`<b>`<c>foo</b>`<bar</c>`</a>

*No, the first b element does not have an end tag!*

g) `<a/>`<b>foo</b>`<b>bar</b>`

*No, this document has no root.*

h) `<a>`<b>`<c>foo</c>`baz`<c>`bar</c>`</b>`</a>

```
  +---+                     +---+
  |   |                     |   |
  +---+                     +---+
      |                     |   |
      |                     | x |
      |                     | @ |
      +---+                   +---+
            +---+               +---+
            |   |               |   |
            +---+               +---+
                |               |   |
                +---+           |   |
                    foo       x 1
                +---+           |   |
                    2             +---+
```

i) `<a x="1" y="2" x="3">`<b>foo</b>`</a>

*No, elements can’t have duplicate attributes, and this has two x’s*
2. 20 pts

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
  <rdfs:Class rdf:ID="Person" />  
  <rdfs:Class rdf:ID="Document" />  
  <rdfs:Class rdf:ID="Book">
    <rdfs:subClassOf rdf:resource="#Document" />  
  </rdfs:Class>
  <Person rdf:ID="king">
    <name>Stephen King</name>
    <wrote rdf:resource="#stand" />  
  </Person>
  <Book rdf:ID="stand">
    <title>The Stand</title>
    <year>1978</year>
  </Book>
</rdf:RDF>
```

-1 if added Property triples for name, wrote, year, title.
-3 if missing <rdf:RDF> tag with rdf and rdfs namespaces
-1 if missing </rdf:RDF> tag
3. 30 pts.

24 total edges: 1 pt each
-1 if no "u" prefix
4. (15 pts.) New triples added: (1.5 pts. each)

- u:sarah rdf:type u:Student (by rdfs3)
- u:cs100 rdf:type u:Course (by rdfs3)
- u:cs200 rdf:type u:Course (by rdfs3)
- u:alan u:knows u:rob (by rdfs7)
- u:alan u:knows u:sarah (by rdfs7)
- u:alan rdf:type u:Person (by rdfs9)
- u:rob rdf:type u:Person (by rdfs9)
- u:sarah rdf:type u:Person (by rdfs9)

Others given by a rule but already in document

- u:rob rdf:type u:Student (by rdfs2)
- u:alan rdf:type u:Professor (by rdfs2)
- u:rob rdf:type u:Student (by rdfs3)

5. (10 pts.) Intersection semantics for rdfs:range make it possible to infer the rdf:type of resources that participate as the object in triples using the property. If there are multiple rdfs:range statements, then each one implies a new type for the resource (since the type of the resource must be of the intersection of all ranges). If new rdfs:range information is discovered, there is no need to retract information. On the other hand, if union semantics were used, one cannot infer anything. Even if only a single range triple was known for the property, it may be possible that there are unknown range triples, and any resource that was an object of a triple using the property could be of the type of one of the unknown ranges, instead of the known one.