
Examen Partiel : XML

Enseignant : Benoît Valiron
10 novembre 2009

Informations

Cet examen partiel contient 6 questions, réparties sur 3 pages. Il est noté sur 20 points. Vous avez droit à vos documents. Vous avez 3 heures (4 si vous bénéficiez du tiers temps).

Nom

Prénom

Numéro d'étudiant

1. Constitution de DTD

6 pts

On veut encoder un arbre généalogique dans un format XML. La racine est `gene`, elle a pour fils des éléments `personne`. On autorise une généalogie vide.

L'élément `personne` a un attribut facultatif `id` de type ID et un attribut obligatoire `genre` pouvant prendre au choix les valeurs `h` et `f`. Il a un fils `nom`, obligatoire, contenant uniquement du texte et un fils `parents`, facultatif, vide. L'ordre est important. L'élément `parents` possède un attribut `ref`, de type IDREFS, obligatoire.

Écrivez la DTD correspondant à cette description, et insérez-la en tant que DTD interne d'un document XML décrivant la généalogie suivante: Bob a deux parents, Marie et Henry. Henry a deux parents, Jean et Françoise. Dans l'arbre XML, vous pouvez utiliser `p` en place de `personne`.

CORRECTION

```
<!DOCTYPE gene [  
  
  <!ELEMENT gene (personne*)>  
  <!ELEMENT personne (nom,parents?)>  
  <!ATTLIST personne id ID #IMPLIED  
                    genre (h|f) #REQUIRED>  
  <!ELEMENT parents EMPTY>  
  <!ATTLIST parents ref IDREFS #REQUIRED>  
  <!ELEMENT nom (#PCDATA)>  
  ]>  
<gene>  
  <personne id="b" genre="h">  
    <nom>Bob</nom>  
    <parents ref="m h" />  
  </personne>  
  <personne id="m" genre="f">  
    <nom>Marie</nom>  
  </personne>  
  <personne id="h" genre="h">  
    <nom>Henry</nom>  
    <parents ref="j f" />  
  </personne>  
  <personne id="j" genre="h">  
    <nom>Jean</nom>  
  </personne>  
  <personne id="f" genre="f">
```

```

    <nom>Françoise</nom>
  </personne>
</gene>

```

2. Document XML bien formé

2 pts

Voilà un document XML avec 5 erreurs. Entourez-les puis corrigez-les (Ne réécrivez pas le document, faites les modifications dans le texte).

```

<?xml version="1.0" encoding="utf-8"?>
<bibliographie>
  <TITRE>Calcul Quantique</titre>
  <article type=non publié>
    <titre>Quantum Proofs for Classical Theorem</titre>
    <auteurs>
      <nom>Drucker</nom><nom>de Wolf</nom>
    </auteurs>
    <uri>arxiv:quant-ph/0910.3376v1</uri>
  </article>
  <livre xml:lang = "en">
    <titre>Quantum Computation and Quantum Information</titre>
    <auteurs>
      <nom>Nielsen</nom><nom>Chuang</nom>
    <éditeur>Cambridge U. Press</éditeur>
  </livre/>
</bibliographie>

```

CORRECTION

Le document corrigé est le suivant :

```

<?xml version="1.0" encoding="utf-8"?>
<bibliographie>
  <titre>Calcul Quantique</titre>
  <article type="non publié">
    <titre>Quantum Proofs for Classical Theorem</titre>
    <auteurs>
      <nom>Drucker</nom><nom>de Wolf</nom>
    </auteurs>
    <uri>arxiv:quant-ph/0910.3376v1</uri>
  </article>
  <livre xml:lang = "en">
    <titre>Quantum Computation and Quantum Information</titre?>
    <auteurs>
      <nom>Nielsen</nom><nom>Chuang</nom>
    </auteurs>
    <éditeur>Cambridge U. Press</éditeur>
  </livre>
</bibliographie>

```

3. Espaces de noms

3 pts

Dans le document suivant, donnez les espaces de noms des éléments

racine aucun abc 4 xyz 3 qwerty 4

poiuy aucun def 2 figure 1 levain 2

Vous pouvez vous contenter d'indiquer les numéros correspondant, ou "aucun" si un élément est sans espace de noms. Écrivez votre réponse dans le blanc laissé à côté de chaque nom d'élément.

```
<?xml version="1.0" encoding="utf-8"?>
<racine xmlns:pre="http://nom-1">
  <pre:abc xmlns:pre="http://nom-4">
    <bio:xyz xmlns:bio="http://nom-3">
      <pre:qwerty />
    <poiuy />
  </bio:xyz>
</pre:abc>
<def xmlns="ftp://nom-2">
  <pre:figure>
    <levain />
  </pre:figure>
</def>
</racine>
```

4. DTD vers Relax NG

6 pts

Voilà une DTD.

```
<!ELEMENT tour (étage)>
<!ATTLIST tour xmlns CDATA #FIXED "http://schema-de-tour">
<!ELEMENT étage (étage|sommet)>
<!ELEMENT sommet EMPTY>
<!ATTLIST étage fenêtres CDATA #REQUIRED>
```

1. Si la racine est `tour`, donnez un schéma Relax NG en format compact qui décrit la même chose, en utilisant le préfixe `t` pour l'espace de nom `http://schema-de-tour` (que vous pouvez abréger en `tour`).

CORRECTION

```
namespace t = "http://schema-de-tour"

start = element t:tour {
  ref_tour
}

ref_tour = (
  element t:étage {
    attribute fenêtres { text },
    ref_tour
  } |
  element t:sommet { empty } ?
)
```

2. Écrivez ce schéma en format XML. N'oubliez pas l'espace de noms de Relax NG : `http://relaxng.org/ns/structure/1.0` (que vous pouvez abréger en `relax`).

CORRECTION

```
<grammar xmlns:t="http://schema-de-tour"
  xmlns="http://relaxng.org/ns/structure/1.0">
```

```

<start>
  <element name="t:tour">
    <ref name="ref_tour"/>
  </element>
</start>
<define name="ref_tour">
  <choice>
    <element name="t:étage">
      <attribute name="fenêtres"/>
      <ref name="ref_tour"/>
    </element>
    <optional>
      <element name="t:sommet">
        <empty/>
      </element>
    </optional>
  </choice>
</define>
</grammar>

```

3. Dans le schéma *compact*, comment faire pour que

- le type de l'attribut *fenêtre* soit `nonNegativeInteger`, de la librairie de type XML Schema (on rappelle que le préfixe standard est `xsd`).
- il y ait une contrainte sur l'attribut *fenêtre* : On limite sa valeur aux nombres compris entre 0 et 10. On pourra utiliser la facette `maxInclusive`.

Indiquez la ligne qui change et les modifications à effectuer.

CORRECTION

```

namespace t = "http://schema-de-tour"

start = element t:tour {
  ref_tour
}

ref_tour = (
  element t:étage {
    attribute fenêtres {
      xsd:nonNegativeInteger {
        maxInclusive='10'
      }
    },
    ref_tour
  } |
  element t:sommet { empty } ?
)

```

5. structure d'un document XML

2 pts

Dans cette partie, écrivez votre réponse à côté des questions.

1. Considérez le document XML suivant :

```
<svg viewBox="0 0 100 100">
```

```
<text x="10" y="90">Bonjour</text>
<rect x="30" y="30" width="40" height="40" stroke="blue"
      fill="green" />
<circle cx="50" cy="30" r="20" stroke="blue" fill="red" />
</svg>
```

Combien y a-t-il d'éléments ? d'attributs ? de noeuds de type texte ?

CORRECTION

Il y a 4 éléments, 14 attributs, 1 noeud de type texte (5 en comptant les espaces).

2. Considérez le document XML suivant :

```
<math><mfrac><msqrt><mi>x</mi></msqrt><mn>2</mn></mfrac></math>
```

Combien y a-t-il d'éléments ? d'attributs ? de noeuds de type texte ?

CORRECTION

Il y a 5 éléments, 0 attribut, 2 noeuds de type texte.

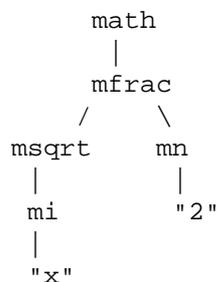
Quels sont les fils de `mfrac` ? Quel est le père de `mi` ? Le père de `mfrac` ?

CORRECTION

Fils de `mfrac` : `msqrt` et `mn`. Père de `mi` : `msqrt`. Père de `mfrac` : `math`.

Représentez le document sous la forme d'un arbre. Quelle est la formule décrite ?

CORRECTION



La formule est "racine de x, le tout sur 2"

6. Entités générales

1 pts

Voila un document XML avec DTD interne.

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE texte [
<!ELEMENT texte (#PCDATA)>
]>
<texte>
  Evilard est une commune suisse du canton de Berne.
</texte>
```

Transformez ce document pour avoir le mot "Evilard" dans une entité générale que vous appellerez ville. Indiquez juste les lignes que vous ajoutez ou modifiez.

CORRECTION

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE texte [
<!ELEMENT texte (#PCDATA)>
<!ENTITY ville "Evilard">
]>
<texte>
  &ville; est une commune suisse du canton de Berne.
</texte>
```