

Vous rendrez vos réponses sur les copies anonymes que l'on vous a distribuées. Les exercices peuvent être traités dans le désordre. Le barème n'est donné qu'à titre indicatif et pourra être modifié.

Dans l'ensemble du sujet, nous considérons des documents XML permettant de décrire l'organisation interne d'une entreprise. Un tel document est organisé suivant les règles suivantes :

- la balise racine est **organisation**
- la balise racine possède deux éléments fils, **services** et **employes**, dans cet ordre
- la balise **services** contient un élément fils nommé **service** pour chaque service de l'entreprise décrit dans le document
- la balise **employes** contient un élément fils nommé **employe** pour chaque employé de l'entreprise décrit dans le document
- chaque service est décrit de la façon suivante :
  - il possède un attribut nommé **num** lui donnant un identifiant
  - il contient un élément fils nommé **nom** contenant son nom
  - il peut contenir un élément fils vide nommé **responsable** possédant un attribut nommé **nom**, et référençant un employé.
- chaque employé est décrit de la façon suivante :
  - il possède un attribut **id** lui donnant un identifiant
  - il possède un élément **nom** contenant son nom
  - il possède (éventuellement) un élément **prenom** contenant son prénom
  - il possède un élément **activites** qui décrit les services dans lesquels il intervient. Cet élément contient des éléments **activite**, qui sont vides, mais possèdent un attribut nommé **ref**, et qui fait référence à un élément **service** défini ailleurs
  - il peut posséder un élément **details**, optionnel, qui peut contenir les informations suivantes (chacune de ces informations est optionnelle) :
    - un élément **age** contenant l'âge
    - un élément **sexe** contenant le sexe

Un exemple (simple) de document respectant ces contraintes est donné dans l'annexe page 5 par le document 1.

### Exercice 1 : DTDs (2 points)

Donnez une DTD décrivant les documents XML utilisés pour l'organisation d'une entreprise. Le document 1 doit être valide par rapport à votre DTD, et de plus elle doit vérifier les critères énoncés dans l'introduction.

### Exercice 2 : Requêtes XPath et XQuery (7,5 points)

Nous considérons toujours les documents XML utilisés pour l'organisation d'une entre-

prise. Les requêtes que vous écrirez par la suite doivent naturellement être valables pour n'importe quel document XML suivant la spécification décrite dans l'introduction.

1. Requêtes XPath (2,5 points = 5×0,5 points)

Donnez des requêtes XPath pour sélectionner les éléments suivants. Décrivez (en français) le contenu du résultat de chaque requête pour le document 1 sans donner tout le code XML renvoyé.

- (a) les employés dont le nom de famille est "Willey".
- (b) les noms de famille des employés dont l'âge est supérieur ou égal à 50 ans.
- (c) les noms de famille des employés travaillant dans le service de comptabilité (en sachant que l'identifiant de ce service est "compta").
- (d) les identifiants des employés travaillant dans au moins deux services.
- (e) compter le nombre de personnes impliquées dans le service de direction.

2. Documents XQuery (5 points)

Pour chacune des questions suivantes, donnez une requête XQuery permettant de construire le document demandé.

- (a) On souhaite modifier légèrement le document XML : nous voulons conserver la partie "services" inchangée, et dans la partie "employés", modifier la façon dont les noms des employés est stockée. Nous voulons fusionner les nom et prénom, en mettant d'abord le prénom, et en plaçant le tout dans une balise identité. Par exemple, pour les deux premiers employés de l'exemple, nous devons obtenir :

```
<employe id="ralf">
  <identite>Ralf Krecker</identite>
  <activites>
    <activite ref="compta"/>
  </activites>
  <details>
    <age>29</age>
    <sexe>Homme</sexe>
  </details>
</employe>
<employe id="joe">
  <identite>Joye Willey</identite>
  <activites>
    <activite ref="compta"/>
    <activite ref="direction"/>
  </activites>
</employe>
```

- (b) Nous supposons que le document est à nouveau dans son état initial. Nous voulons à présent l'aplatir, c'est-à dire qu'il ne contienne que la partie **services**, mais dans laquelle on donne explicitement les employés dans chaque service. De plus,

dans cette question, on ne conserve pas l'information concernant le responsable. Un élément **service** contiendra donc une balise **nom** puis une balise **employes**, qui contiendra tous les employés effectivement affectés à ce service. Ici, les éléments **employe** ne contiendront donc plus de balise **activites**. Ils ne contiendront plus non plus d'identifiant. Vous pouvez pour réaliser cela définir des fonctions, par exemple une fonction recopiant un employé en enlevant la partie **activites**. Par exemple, pour le premier service, on doit obtenir :

```
<service num="direction">
  <nom>Direction</nom>
  <employes>
    <employe>
      <nom>Willey</nom>
      <prenom>Joey</prenom>
    </employe>
    <employe>
      <nom>Nick</nom>
      <details>
        <age>35</age>
      </details>
    </employe>
  </employes>
</service>
```

- (c) Modifiez la requête construite à la question précédente pour conserver l'information concernant le responsable du service. Pour cela, on ajoutera à l'employé responsable du service un élément vide **responsable**. Par exemple pour le premier service, on doit obtenir :

```
<service num="direction">
  <nom>Direction</nom>
  <employes>
    <employe>
      <nom>Willey</nom>
      <prenom>Joey</prenom>
      <responsable/>
    </employe>
    <employe>
      <nom>Nick</nom>
      <details>
        <age>35</age>
      </details>
    </employe>
  </employes>
</service>
```

**Exercice 3 : Transformations XSLT (5 points)**

**Affichage XSL (2,5 points)** Proposez une requête XQuery permettant de construire la page web (X)HTML dont un extrait est représenté dans le document 2 page 6. Bien sûr, votre transformation doit pouvoir être appliquée à n'importe quel document XML du même type que celui considéré en exemple dans ce sujet. Notez la présentation des détails des employés, des responsabilités éventuelles...

**Production XSL-FO (2,5 points)** Ecrivez un document XSLT permettant la production d'un document XSL-FO pour réaliser la présentation de l'organisation de l'entreprise. Ce catalogue présentera un contenu proche de celui de la page web réalisée précédemment. Plus précisément, on isolera une première page contenant le titre du document, la liste des services de l'entreprise et le nombre d'employés. Les pages suivantes présenteront les employés un par un, comme cela est fait dans la page web.

**Exercice 4 : APIs Java (2 points)**

Ecrivez une extension de la classe SimpleSaxParser permettant d'afficher les noms de famille des employés de l'entreprise travaillant pour le service "Comptabilité". Attention, vous ne connaissez pas a priori l'identifiant de ce service.

**Exercice 5 : Automates d'arbres (3,5 points)**

**DTDs.** Nous considérons les deux DTD suivantes :

DTD 1 :

```
<!ELEMENT gene (dtype|rtype)*>
<!ELEMENT rtype (#PCDATA)>
<!ELEMENT dtype (age,year,size)>
<!ELEMENT age (#PCDATA)>
<!ELEMENT year (#PCDATA)>
<!ELEMENT size (#PCDATA)>
```

DTD 2 :

```
<!ELEMENT spec (dtype|rtype)*>
<!ELEMENT rtype (#PCDATA)>
<!ELEMENT dtype (year,size)>
<!ELEMENT year (#PCDATA)>
<!ELEMENT size (#PCDATA)>
```

1. Pour chaque DTD, donnez un automate d'arbre descendant déterministe associé à cette DTD.
2. Donnez un automate descendant déterministe reconnaissant l'union des deux DTDs.
3. Existe-t-il une DTD permettant de décrire cette union ? Expliquez pourquoi.

**Automates.** Nous considérons les arbres binaires (tout noeud a exactement 0 ou 2 fils) construits sur les deux symboles  $a$  et  $b$ . Donnez un automate d'arbre ascendant déterministe permettant de reconnaître les arbres binaires de cette nature dont le nombre de noeuds étiquetés par un  $a$  est pair.

Vous expliquerez avec précision les différents éléments permettant de construire votre automate.

# Annexes

## Document 1 : Un document de gestion d'entreprise

```
<organisation>
  <services>
    <service num="direction">
      <nom>Direction</nom>
      <responsable nom="joe"/>
    </service>
    <service num="compta">
      <nom>Comptabilite</nom>
    </service>
    <service num="com">
      <nom>Communication</nom>
      <responsable nom="nick"/>
    </service>
  </services>
  <employes>
    <employe id="ralf">
      <nom>Krecker</nom>
      <prenom>Ralf</prenom>
      <activites>
        <activite ref="compta"/>
      </activites>
      <details>
        <age>29</age>
        <sexe>Homme</sexe>
      </details>
    </employe>
    <employe id="joe">
      <nom>Willey</nom>
      <prenom>Joey</prenom>
      <activites>
        <activite ref="compta"/>
        <activite ref="direction"/>
      </activites>
    </employe>
    <employe id="nick">
      <nom>Nick</nom>
      <activites>
        <activite ref="com"/>
        <activite ref="direction"/>
      </activites>
    </employe>
  </employes>
</organisation>
```

```

    </activites>
    <details>
      <age>35</age>
    </details>
  </employe>
</employes>
</organisation>

```

## Document 2 : Présentation HTML de l'entreprise

```

<html>
  <head>
    <title>Organisation de l'entreprise</title>
  </head>
  <body>
    Les employés sont
    <ul>
      <li><p>
        Ralf Krecker, Homme de 29 ans.
        <ul>
          <li>Comptabilité</li>
        </ul>
      </p></li>
      <li><p>
        Joey Willey.
        <ul>
          <li>Comptabilité</li>
          <li>Direction, responsable</li>
        </ul>
      </p></li>
      <li><p>
        Nick, 35 ans.
        <ul>
          <li>Communication, responsable</li>
          <li>Direction</li>
        </ul>
      </p></li>
    </ul>
  </body>
</html>

```