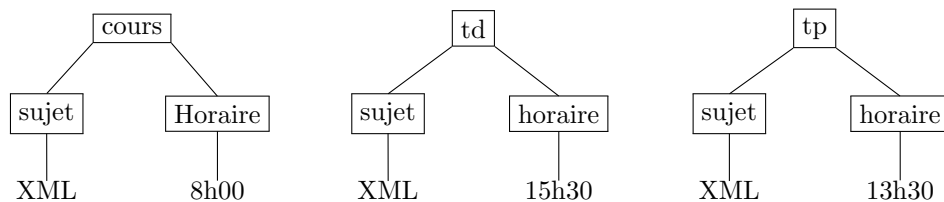


Exercice 1.

Question 1. Les 6 erreurs sont :

1. Il n'y a pas de balise racine.
2. Les attributs *version* et *encoding* de la balise xml d'en-tête doivent apparaitre dans cette ordre (*version* puis *encoding*).
3. La balise du prologue est mal fermée.
4. Il y a une différence de nommage entre la balise d'ouverture et de fermeture sur la première balise *Horaire*.
5. La balise *td* n'est pas fermée.
6. Il y a une différence de nommage entre la balise d'ouverture et de fermeture de la balise *tp jour*.

Question 2. L'arbre correspondant :



Exercice 2.

- `<123>` : Invalide car ne commence pas par une lettre.
- `<.aaa>` : Invalide car commence par un point.
- `<dd;ee>` : Invalide car contient un point virgule.
- `<1qwerty>` : Invalide car commence par un chiffre.
- `<aa __>` : Invalide car contient un espace.
- `<xmltest>` : Invalide car commence par *xml*.

Exercice 3.

Question 1. Les fichiers `a1.xml` et `a2.xml` sont valides par rapport à leurs DTD respectifs. Par contre `a3.xml` ne l'est pas car la balise A est de la forme (B D B D) alors que le DTD décrit (B+, D+, F*).

Question 2.

a1.xml

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<!DOCTYPE A SYSTEM "d1.dtd">
```

```

<A>
  <B>
    <C>1</C>
    <C>2</C>
  </B>

  <F>3</F>
</A>

```

a2.xml

```

<?xml version="1.0" encoding="utf-8"?>

<!DOCTYPE A SYSTEM "d2.dtd">

<A>
  <B>
    <C>4</C>
  </B>

  <D>
    <F>5</F>
    <F>6</F>
  </D>

  <F>7</F>
</A>

```

a3.xml

```

<?xml version="1.0" encoding="utf-8"?>

<!DOCTYPE A SYSTEM "d3.dtd">

<A>
  <B>
    <C>8</C>
  </B>

  <D>
    <F>9</F>
    <F>10</F>
  </D>

  <B></B>

  <D>
    <F>11</F>
  </D>
</A>

```

Exercise 4.

Question 1.

movies.xml

```

<?xml version="1.0" encoding="utf-8" standalone="no"?>

```

```

<!DOCTYPE movies SYSTEM "exo4-dtd.dtd">

<movies>
  <movie id="TBCB">
    <title>The Broken Circle Breakdown</title>
    <year>2012</year>
    <runtime>111m</runtime>
    <directors>
      <director>
        <name>
          <firstname>Felix</firstname>
          <lastname>Van Groeningen</lastname>
        </name>
      </director>
    </directors>
    <actor>
      <name>
        <firstname>Johan</firstname>
        <lastname>Heldenbergh</lastname>
      </name>
      <role>Didier Bontinck</role>
    </actor>
    <actor>
      <name>
        <firstname>Veerle</firstname>
        <lastname>Baetens</lastname>
      </name>
      <role>Elise Vandeveld</role>
    </actor>
    <summary>One of the best movie ever.</summary>
  </movie>
</movies>

```

Question 2.

```
<!ELEMENT name ((firstname, lastname) | (lastname, firstname))>
```

Question 3.

Il faudrait suivre la même logique que pour la question précédente et écrire les $6! = 720$ possibilités d'ordres.

Exercise 5.

Question 1.

Question 2.

Question 3.

exo5-schema.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="personnes">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="personne" minOccurs="0" maxOccurs="
          unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

```

</xs:element>
<xs:element name="personne">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="nom" type="xs:string"/>
      <xs:element name="prenom" type="xs:string"/>
      <xs:element name="naissance" type="type-evenement"/>
      <xs:element name="deces" type="type-evenement" minOccurs="0"
        maxOccurs="1"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element>
<xs:complexType name="type-evenement">
  <xs:sequence>
    <xs:element name="date" type="xs:date"/>
    <xs:element name="lieu" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
</xs:schema>

```