

Bases de Données II, Charleroi, 11 janvier 2019

NOM + PRÉNOM :

Orientation + Année :

Cet examen contient 10 questions.

On utilise un document XML pour encoder les stocks de bières en bouteille disponibles auprès de différents distributeurs (Carrefour, Colruyt, Delhaize...). Les bières sont regroupées par brasserie (Brasserie de la Senne, Brasserie d'Achouffe, Chimay...). Chaque brasserie est située dans une seule ville (attribut `ville`). Chaque bière porte un nom unique (attribut `marque`); le degré alcoolique est exprimé en pourcentage volumique (attribut `alcool`). Les bières sont disponibles en bouteilles de 25, 33, 37.5, 75 ou 100 centilitres. Certaines bières sont disponibles en plusieurs bouteilles; par exemple, la Chimay Bleue est disponible en bouteilles de 33 et 75 centilitres. Carrefour stocke 3000 bouteilles de 33 centilitres de Chimay Bleue.

La DTD est incluse au début du document XML de la figure 1.

Pour les questions 1 à 4, évitez, si possible, l'usage des axes *parent* et *ancestor*.

Question 1 Écrivez une expression XPath qui rend les noms des brasseries qui produisent au moins une bière dans une bouteille de 75 centilitres. Pour le document de la figure 1, la réponse est comme suit :

```
nom="Chimay"
```

```
nom="Brasserie d'Achouffe"
```

.../5

Question 2 Écrivez une expression XPath qui rend le nom de la brasserie qui produit la bière la plus forte. Pour le document de la figure 1, la réponse est comme suit :

```
nom="Chimay"
```

Pour comprendre cette réponse, noter que la bière la plus forte est la Chimay Bleue produite par la brasserie Chimay.

.../5

Question 3 Écrivez une expression XPath qui rend le distributeur stockant le plus grand nombre de bouteilles de 33 centilitres de Chimay Rouge. Pour cette question, il n'est pas permis d'utiliser des fonctions d'agrégation, telles que count, max, min... Pour le document de la figure 1, la réponse est comme suit :

```
distributeur="Colruyt"
```

.../5

Question 4 Écrivez une expression XPath qui rend les noms des brasseries qui coopèrent avec deux ou plusieurs distributeurs. Pour cette question, il n'est pas permis d'utiliser des fonctions d'agrégation, telles que count, max, min... Il n'est pas non plus permis d'utiliser les axes *following-sibling*, *preceding-sibling*, *following* ou *preceding*. Pour le document de la figure 1, la réponse est comme suit :

```
nom="Chimay"  
nom="Brasserie d'Achouffe"
```

La brasserie Timmermans ne fait pas partie de la réponse car ses bières sont distribuées exclusivement par Colruyt.

Astuce : Soit A un ensemble. Selon la sémantique existentielle, $A \neq A$ est vrai si et seulement si A contient au moins deux éléments.

.../5

```

<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE STOCKS [
<!ELEMENT STOCKS (brasserie)*>
<!ELEMENT brasserie (biere)*>
<!ELEMENT biere (stock)*>
<!ELEMENT stock (#PCDATA)>
<!ATTLIST brasserie nom CDATA #REQUIRED>
<!ATTLIST brasserie ville CDATA #REQUIRED>
<!ATTLIST biere marque CDATA #REQUIRED>
<!ATTLIST biere alcool CDATA #REQUIRED>
<!ATTLIST stock distributeur CDATA #REQUIRED>
<!ATTLIST stock bouteille (25|33|37.5|75|100) #REQUIRED>
]>
<STOCKS>
  <brasserie nom="Chimay" ville="Scourmont">
    <biere marque="Chimay Bleue" alcool="9">
      <stock distributeur="Carrefour" bouteille="33">3000</stock>
      <stock distributeur="Carrefour" bouteille="75">1000</stock>
      <stock distributeur="Colruyt" bouteille="33">1000</stock> </biere>
    <biere marque="Chimay Rouge" alcool="7">
      <stock distributeur="Carrefour" bouteille="33">2500</stock>
      <stock distributeur="Carrefour" bouteille="75">2000</stock>
      <stock distributeur="Colruyt" bouteille="33">2600</stock>
      <stock distributeur="Delhaize" bouteille="33">2500</stock> </biere>
  </brasserie>
  <brasserie nom="Brasserie d'Achouffe" ville="Achouffe">
    <biere marque="La Chouffe" alcool="8">
      <stock distributeur="Carrefour" bouteille="33">1000</stock>
      <stock distributeur="Carrefour" bouteille="75">100</stock>
      <stock distributeur="Carrefour" bouteille="100">10</stock>
      <stock distributeur="Colruyt" bouteille="33">1000</stock> </biere>
    <biere marque="Chouffe Bok 6666" alcool="6.66">
      <stock distributeur="Carrefour" bouteille="33">500</stock>
      <stock distributeur="Colruyt" bouteille="33">500</stock> </biere>
    <biere marque="Chouffe Soleil" alcool="6">
      <stock distributeur="Carrefour" bouteille="33">250</stock>
      <stock distributeur="Colruyt" bouteille="33">250</stock> </biere>
  </brasserie>
  <brasserie nom="Timmermans" ville="Bruxelles">
    <biere marque="Timmermans Framboise" alcool="4">
      <stock distributeur="Colruyt" bouteille="25">900</stock>
      <stock distributeur="Colruyt" bouteille="33">700</stock> </biere>
    <biere marque="Timmermans Kriek Retro" alcool="5">
      <stock distributeur="Colruyt" bouteille="37.5">250</stock> </biere>
  </brasserie>
  <brasserie nom="Brasserie de la Senne" ville="Bruxelles">
    <biere marque="Stouterik" alcool="4.5">
      <stock distributeur="Colruyt" bouteille="33">500</stock> </biere>
    <biere marque="Brusseleir" alcool="8">
      <stock distributeur="Colruyt" bouteille="33">400</stock> </biere>
    <biere marque="Zinnebir" alcool="6">
      <stock distributeur="Colruyt" bouteille="33">400</stock> </biere>
  </brasserie>
</STOCKS>

```

FIGURE 1 – Stocks de bières.

```

<distributeurs>
  <Delhaize>
    <DistributeurPour>Chimay</DistributeurPour>
  </Delhaize>
  <Carrefour>
    <DistributeurPour>Chimay</DistributeurPour>
    <DistributeurPour>Brasserie d'Achouffe</DistributeurPour>
  </Carrefour>
  <Colruyt>
    <DistributeurPour>Chimay</DistributeurPour>
    <DistributeurPour>Brasserie d'Achouffe</DistributeurPour>
    <DistributeurPour>Timmermans</DistributeurPour>
    <DistributeurPour>Brasserie de la Senne</DistributeurPour>
  </Colruyt>
</distributeurs>

```

FIGURE 2 – Output du programme XSLT de la question 5.

```

<BOUTEILLES>
  <BOUTEILLE-75>
    <biere marque="Chimay Bleue"/>
    <biere marque="Chimay Rouge"/>
    <biere marque="La Chouffe"/>
  </BOUTEILLE-75>
  <BOUTEILLE-100>
    <biere marque="La Chouffe"/>
  </BOUTEILLE-100>
  <BOUTEILLE-25>
    <biere marque="Timmermans Framboise"/>
  </BOUTEILLE-25>
  <BOUTEILLE-37.5>
    <biere marque="Timmermans Kriek Retro"/>
  </BOUTEILLE-37.5>
  <BOUTEILLE-33>
    <biere marque="Chimay Bleue"/>
    <biere marque="Chimay Rouge"/>
    <biere marque="La Chouffe"/>
    <biere marque="Chouffe Bok 6666"/>
    <biere marque="Chouffe Soleil"/>
    <biere marque="Timmermans Framboise"/>
    <biere marque="Stouterik"/>
    <biere marque="Brusseleir"/>
    <biere marque="Zinnebir"/>
  </BOUTEILLE-33>
</BOUTEILLES>

```

FIGURE 3 – Output du programme XSLT de la question 6.

Question 5 Dans le programme XSLT ci-après, quatre expressions ont été remplacées par les chaînes XXX111, XXX222, XXX333 et XXX444. Le programme sert à générer un document XML affichant une (et une seule) fois chaque distributeur. Chaque distributeur est suivi par la liste (sans doublons!) des brasseries pour lesquelles il distribue une ou plusieurs bières, dans le format illustré par la figure 2.

Donnez les valeurs originales des quatres chaînes.

.../10

```
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">
  <distributeurs>
    <xsl:apply-templates select="XXX111"/>
  </distributeurs>
</xsl:template>

<xsl:template match="@distributeur">
  <xsl:element name="XXX222">
    <xsl:apply-templates select="XXX333"/>
  </xsl:element>
</xsl:template>

<xsl:template match="brasserie">
  <xsl:element name="DistributeurPour">
    <xsl:value-of select="XXX444"/>
  </xsl:element>
</xsl:template>

</xsl:stylesheet>
```

Les valeurs à substituer :

XXX111=

XXX222=

XXX333=

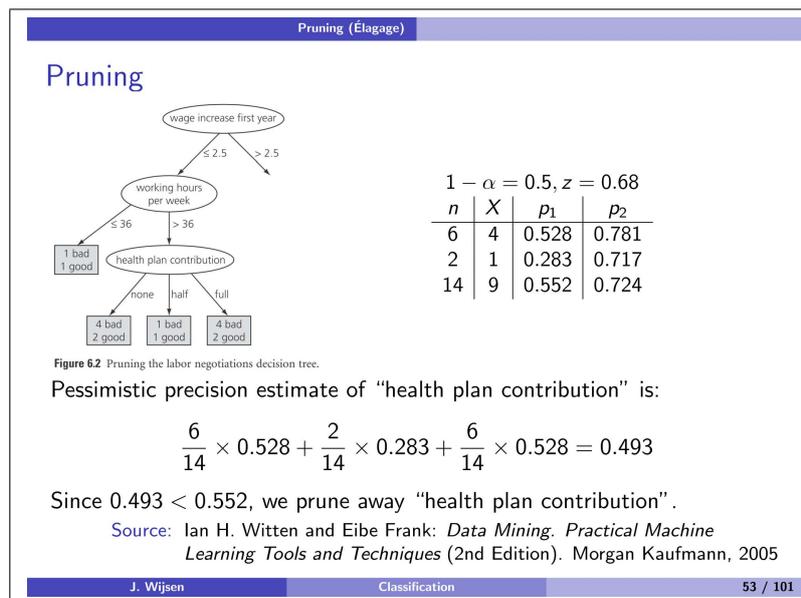
XXX444=

Question 6 Écrivez un programme XSLT qui affiche les différentes tailles de bouteille. Chaque taille de bouteille est suivie par la liste des bières disponibles dans cette taille dans le format illustré par la figure 3. **LES CONSTANTES 25, 33, 37.5, 75 ou 100 NE PEUVENT PAS APPARAÎTRE DANS VOTRE PROGRAMME**; les tailles doivent donc être récupérées par une requête XPath. L'ordre dans lequel les tailles de bouteille sont affichées n'a pas d'importance.

La position des blancs et retours à la ligne n'a pas d'importance. Le programme ne peut pas contenir les balises `xsl:for-each`, `xsl:if` ou `xsl:with-param`.

.../10

Question 7 Expliquez en détail l'image suivante.



Vous devez expliquer l'intérêt général de *pruning*, mais aussi la façon d'obtenir les nombres concrets dans l'image.

.../20

Question 8 Expliquez en détail l'image suivante.

FP-tree : une structure de données pour calculer les intersections en RAM (Random Access Memory)

Transaction Data Set	
TID	Items
1	{a,b}
2	{b,c,d}
3	{a,c,d,e}
4	{a,d,e}
5	{a,b,c}
6	{a,b,c,d}
7	{a}
8	{a,b,c}
9	{a,b,d}
10	{b,c,e}

Parmi les 8 transactions qui contiennent *a*, il y en a 5 qui contiennent aussi *b*. Parmi ces 5 transactions qui incluent *ab*, il y en a 3 qui contiennent aussi *c*. Etc.

J. Wijsen Association Rules 20 / 21

Vous devez détailler

1. le problème qui est à la base de cette image ;
2. comment l'*FP-tree* peut contribuer à la résolution de ce problème ;
3. les étapes du calcul qui suivent la construction de l'*FP-tree*. Illustrez ces étapes à l'aide de cet exemple concret ;
4. d'autres éléments que vous jugez pertinents.

.../20

Question 9 Détaillez le fonctionnement de la méthode de *cross-validation* dans la classification et expliquez l'objectif de cette méthode.

.../10

Question 10 Situez chaque terme dans le cursus et expliquez de façon succincte mais précise.

Closed itemset.

.../5

Oblique decision tree.

.../5
