

id21-final

1. Automates

Soit l'alphabet $\Sigma = \{a, b\}$ et A un automate fini selon la notation $A = (Q, q_0, F, \delta)$ où Q est l'ensemble des états, $q_0 \in Q$ est l'état initial, $F \subseteq Q$ est l'ensemble des états finaux et δ est l'ensemble des transitions. Soit $A = (\{1, 2, 3\}, 1, \{3\}, \{(1, a, 2), (2, b, 1), (2, a, 3)\})$. L'automate a 3 états, l'état initial est 1, le seul état final est 3 et il y a 3 transitions. Soit $L(A)$ l'ensemble des mots acceptés par A . Choisir une des possibilités.

- (a) $L(A) = (ab)^*.a$
- (b) $L(A) = ((ab)^*.a + a)$
- (c) $L(A) = ab.(ab)^*.a$
- (d) $L(A) = a.(ba)^*.a$

2. Analyse de mots

Soit l'alphabet $\Sigma = \{a, b\}$ et r une expression régulière qui définit un langage $L(r) \subseteq \Sigma^* = (a + b)^*$. Soit $\overline{L(r)}$ le complément de $L(r)$, l'ensemble des mots de Σ^* qui ne sont pas dans $L(r)$.

Soit r_1 et r_2 deux expressions régulières qui définissent des langages distincts. Choisir une des 4 possibilités.

- (a) L'intersection de $L(r_1)$ et de $L(r_2)$ est un langage régulier et $\overline{L(r_1)}$ n'est pas un langage régulier.
- (b) L'intersection de $L(r_1)$ et de $L(r_2)$ est un langage régulier et $\overline{L(r_1)}$ est un langage régulier.
- (c) L'intersection de $L(r_1)$ et de $L(r_2)$ n'est pas un langage régulier et $\overline{L(r_1)}$ est un langage régulier.
- (d) L'intersection de $L(r_1)$ et de $L(r_2)$ n'est pas un langage régulier et $\overline{L(r_1)}$ n'est pas un langage régulier.

3. Arbres

Soit le fichier .xml suivant (schéma OLAP simplifié):

```
< C >< T >< /T >< D >< H >< L >< /L >< /H >< /D >  
< D >< H >< T >< /T >< L >< /L >< /H >< /D >
```

$\langle M \rangle \langle /M \rangle \langle /C \rangle$

Comment décrire l'arbre associé à ce fichier?

- (a) L'arbre a 8 noeuds et une profondeur 3
- (b) L'arbre a 8 noeuds et une profondeur 4
- (c) L'arbre a 10 noeuds et une profondeur 3
- (d) L'arbre a 10 noeuds et une profondeur 4

4. DTD

Considérons la DTD suivante sur l'alphabet $\Sigma = \{a, b\}$:

$$a : a, b + \Lambda$$
$$b : \Lambda$$

Choisir la bonne réponse concernant un arbre valide pour cette DTD:

- (a) La profondeur est bornée
- (b) Chaque noeud a deux successeurs et la profondeur est arbitraire
- (c) Le degré d'un noeud est arbitraire
- (d) Toutes les branches de l'arbre ont la même profondeur

5. Schéma relationnel

Soit $R_1(A, B, C)$ une relation dont les 3 attributs sont A, B, C et $R_2(A, D)$ une relation dont les 2 attributs sont A, D . Soit la requête SQL:

Select B,C from R_1, R_2 where $R_1.A = R_2.A$

Choisir une des possibilités:

- (a) Le résultat est une relation binaire obtenue par projection de la jointure de R_1 et R_2
- (b) Le résultat est la jointure de R_1 et R_2
- (c) Le résultat est une relation avec 4 attributs
- (d) Le résultat est une relation avec 5 attributs

6. Dépendences

Soit $R(A, B, C)$ une relation dont les 3 attributs sont A, B, C et $T(C, D)$ une relation dont les 2 attributs sont C, D . On note une dépendance fonctionnelle $A \rightarrow B, C$ pour indiquer que la connaissance de A implique la connaissance de B et de C . Supposons aussi que $C \rightarrow D$.

Choisir une des possibilités:

- (a) A est une clé pour R et on n'a pas $A \rightarrow D$
- (b) A n'est pas une clé pour R et on n'a pas $A \rightarrow D$
- (c) A n'est pas une clé pour R et on a $A \rightarrow D$
- (d) A est une clé pour R et on a $A \rightarrow D$

7. OLAP

Soit $R(A, D, E, M)$ une table représentant un entrepôt de données où la mesure est M , une valeur entière. $T(A, B, C)$ est une relation auxiliaire dont les 3 attributs sont A, B, C et A est une clé primaire pour T . Le schéma OLAP a M pour mesure avec la somme pour agrégation et les dimensions discrètes B, C, D . Soit la requête OLAP en SQL étendu:

```
Select sum(M) from R, T where R.A = T.A group by B
```

Choisir une des possibilités:

- (a) Le résultat est un tableau à 2 colonnes et un piechart à deux dimensions
- (b) Le résultat est un tableau à 1 colonne et un piechart à une dimension
- (c) Le résultat est un tableau à 2 colonnes et un piechart à une dimension
- (d) Le résultat est un tableau à 1 colonne et un piechart à deux dimensions

8. Distributions

Soit un Schéma OLAP avec une table entrepôt $R(A, D, E, M)$ où la mesure est M , une valeur entière. $T(A, B, C)$ est une relation auxiliaire dont les 3 attributs sont A, B, C et A est une clé primaire pour T . Sur ce schéma, soit Q la requête OLAP en SQL étendu:

Select sum(M) from R, T where $R.A = T.A$ group by B

Soit $R1$ obtenu comme une table où on a dupliqué R . Chaque tuple de R apparaît 2 fois dans $R1$. On compare la réponse Q_R de la requête Q sur R à la réponse Q_{R1} de la requête Q sur $R1$. Choisir une des possibilités:

- (a) Q_R et Q_{R1} sont les mêmes tableaux et les mêmes piecharts
- (b) Q_R et Q_{R1} sont les mêmes tableaux et des piecharts distincts
- (c) Q_R et Q_{R1} sont des tableaux distincts et les mêmes piecharts
- (d) Q_R et Q_{R1} sont des tableaux distincts et des piecharts distincts

9. Arbres de décision

Soit $T(A, B, C)$ une relation dont les 3 attributs sont A, B, C et le domaine de C est $\{0, 1\}$. Supposons que $A \rightarrow C$ (il existe une dépendance fonctionnelle de A vers C) et que B n'implique pas C . On observe une relation T de 1000 tuples et divise T en deux parties T_1, T_2 de 500 tuples. On construit un arbre de décision à partir de T_1 à l'aide du Gain d'information et on le teste sur T_2 . Choisir une des possibilités:

- (a) L'arbre de décision est de profondeur 1 et l'erreur est nulle
- (b) L'arbre de décision est de profondeur 1 et l'erreur n'est pas nulle
- (c) L'arbre de décision est de profondeur 2 et l'erreur est nulle
- (d) L'arbre de décision est de profondeur 2 et l'erreur n'est pas nulle

10. Clustering

Soit $T(A, B, C)$ une relation dont les 3 attributs sont A, B, C et on applique l'algorithme k -means qui trouve k clusters. Choisir une des possibilités:

- (a) Les clusters forment une partition qui exclut les points extrêmes
- (b) Les clusters forment une partition qui n'exclut aucun point
- (c) Les clusters peuvent se recouper (intersection non nulle) et exclure des points extrêmes
- (d) Les clusters peuvent se recouper (intersection non nulle) et n'excluent aucun point