

--	--	--

Systèmes de Gestion de Bases de Données – 3I009

EXAMEN - 1^{ère} session du 11 janvier 2019

Durée : 2 heures

Documents autorisés

Les téléphones mobiles doivent être éteints et rangés dans les sacs. Le barème sur 21 points (25 questions) n'a qu'une valeur indicative.

1 Questions de cours et de TME (3 pts)

Question 1 (1 point)

Exprimez $T = R(XY) \div S(X)$ la division algébrique de la table $R(XY)$ par la table $S(X)$ en fonction des opérateurs de base de l'algèbre relationnelle. Vous pouvez utiliser des variables pour simplifier l'expression. *Attention, le corrigé de janvier 2018 est erroné.*

Réponse :

Question 2 (1 point)

Démontrer le théorème du cours suivant : « La décomposition de $R(XYZ)$ en $R1(XY)$ et $R2(YZ)$ est SPI si Y est surclé de $R1$ ou de $R2$ (ou des deux) » (on ne démontre pas la partie «et seulement si»)

Réponse :

Question 3 (1 point)

Soit les tables $R1(A,B)$ et $R2(B,C)$. Donner la commande permettant d'effectuer la requête suivante sous Oracle **en utilisant l'algorithme des boucles imbriquées** : $\Pi_A(R1 \bowtie (\sigma_{C=1}(R2)))$

Réponse :

2 Dépendances fonctionnelles (4 pts)

On considère une table $R(A,B,C,D,E,F,G)$ et l'ensemble de dépendances fonctionnelles suivant sur R :

$\mathcal{F} = \{ DFG \rightarrow A; \quad ABD \rightarrow CE; \quad CF \rightarrow A; \quad D \rightarrow A; \quad BE \rightarrow FD; \}$

Question 4 ($\frac{1}{2}$ point)

Donnez deux clés minimales de la table R par rapport à \mathcal{F} .

Réponse :

Question 5 ($\frac{1}{2}$ point)

Montrez que la dépendance $DFG \rightarrow A$ est redondante.

Réponse :

Question 6 ($\frac{1}{2}$ point)

Donnez une couverture minimale de \mathcal{F} .

Réponse :

Question 7 ($\frac{1}{2}$ point)

Est-ce que la décomposition suivante de R est SPI par rapport à \mathcal{F} ? Justifiez votre réponse.

- R1(B, E, F)
- R2(B, D, E)
- R3(A, B, C, D, G)

Réponse :

Question 8 (1 point)

Donnez l'ensemble des dépendances projetées sur chacune des relations R1, R2 et R3. Est-ce que cette décomposition est sans perte de dépendance ? Justifiez votre réponse.

Réponse :

Question 9 ($\frac{1}{2}$ point)

La relation R3 est-elle en 3FN ? Justifiez votre réponse.

Réponse :

Question 10 ($\frac{1}{2}$ point)

Donnez une décomposition SPI et SPD en 3FN de **R** par rapport à \mathcal{F} (ensemble minimal).

Réponse :

3 Optimisation de requêtes (4 pts)

On considère le schéma

Albums (idAlbum, idArtiste, titreA, année, genre)

Chansons (idC, idAlbum, titreC, duréeC)

Artistes (idArtiste, nom)

Ecoutes (idAbonne, date, idC, duréeE)

Abonnés (idAbonne, pseudo)

On connaît les cardinalités et les tailles suivantes (ne pas confondre cardinalité et taille) :

Relation	Cardinalité	Taille en pages
Albums	500	100
Artistes	100	10
Chansons	5 000	20
Ecoutes	1 million	150
Abonnés	10 000	50

On connaît, entre autres, les domaines des attributs suivants :

Relation	Attribut	Nb valeurs	Rmq
Albums	année	50	dans [1968, 2018]
Albums	genre	20	'Rock', ...
Ecoutes	duréeE		de 0 :00 à 10 :00 minutes
Chansons	duréeC		de 0 :00 à 10 :00 minutes

Question 11 (1 point)

On considère un prédicat s_i pour une relation R et le facteur de sélectivité $SF_i = SF(\sigma_{s_i}(R))$. Calculer SF_i .

s_1 : durée BETWEEN 02 :30 AND 06 :30

s_2 : genre='Rock' AND année > 2014

s_3 : (genre = 'Rock' OR genre = 'RnB' OR genre = 'Jazz' OR genre = 'Electro') OR année <= 2008

s_4 : idAbonne = 'ab1' AND idC = 't13'

Réponse :

Pour Ecoutes : $SF_1 =$

Pour Albums : $SF_2 =$

Pour Albums : $SF_3 =$

Pour Ecoutes : $SF_4 =$

Question 12 (1 point)

Quelle est la cardinalité $C_i = \text{card}(R_i)$ des requêtes suivantes ? Ecrire la valeur puis justifier votre réponse.

$R_1 = \text{Chansons} \bowtie_{idC} \text{Ecoutes} \bowtie_{idAbonne} \text{Abonnés}$

$R_2 = (\sigma_{dureeC < 08:00} \text{Chansons}) \bowtie_{idAlbum} (\sigma_{genre='Rock'} \text{Albums})$

$R_3 = (\sigma_{idAlbum='B05EQ'} \text{Chansons}) \bowtie_{idC} \text{Ecoutes}$

$R_4 = (\pi_{idArtiste}(\sigma_{genre='Rock'} \text{Albums})) \bowtie_{idArtiste} (\pi_{idArtiste}(\sigma_{genre='Jazz'} \text{Albums}))$

Réponse :

$C_1 = \dots\dots\dots$ Justif :

$C_2 = \dots\dots\dots$ Justif :

$C_3 = \dots\dots\dots$ Justif :

$C_4 = \dots\dots\dots$ Justif :

Question 13 (1 point)

Pour les questions suivantes, tous les coûts sont exprimés **en nombre de pages lues**. Le coût d'un accès par **index** est $\text{coût}(\sigma_{a \text{ op } x}(R)) = \text{card}(\sigma_{a \text{ op } x}(R))$ tel que *op* est un opérateur (=, <, >) et *x* est une valeur (lire une page pour chaque nuplet accédé à partir des ROWID obtenus en parcourant l'index).

Il y a les six index suivants :

- $\text{Ecoutes}(idAbonne)$, $\text{Ecoutes}(idC)$, $\text{Ecoutes}(duréeE)$
- $\text{Chansons}(idC)$, $\text{Chansons}(idAlbum)$,
- $\text{Albums}(idAlbum)$.

Les jointures sont traitées par **boucles imbriquées avec index**. Pour chaque requête T_i calculer le coût des plans avec index (coût_avec) et sans index (coût_sans) pour la sélection.

$T_1 : \sigma_{idAbonne='ab1'} \text{Ecoutes}$

$T_2 : \sigma_{idC='t11'} \text{Ecoutes}$

$T_3 : (\sigma_{idC='t25'} \text{Chansons}) \bowtie_{idAlbum} \text{Albums}$

$T_4 : (\sigma_{genre='Rock'} \text{Albums}) \bowtie_{idAlbum} \text{Chansons}$

Réponse :

$\text{coût_avec}(T_1) =$ $\text{coût_sans}(T_1) =$

$\text{coût_avec}(T_2) =$ $\text{coût_sans}(T_2) =$

coût_avec(T_3) =coût_sans(T_3) =coût_avec(T_4) =coût_sans(T_4) =**Question 14** (1 point)Soit la requête V :**select** * **from** Artistes a, Albums b**where** a.idArtiste = b.idArtiste **and** b.année = 2018 ;

Proposer un plan pour cette requête qui n'utilise **pas d'index** et évalue la **jointure par hachage**. Le plan choisi minimise le nombre de pages de données servant à construire la table de hachage.

La table de hachage contient-elle des données d'Artistes ou d'Albums ? Quelle est la taille (en nombre de pages disque) des données mises dans la table de hachage ? Quel est le coût de V (en nombre de pages lues) ? Décrire brièvement le plan.

Réponse :

La table de hachage contient des données de la table (entourer) Artistes Albums

Taille en mémoire (exprimée en nombre de pages disque) : pages.

Coût(T_4) pages lues

Brève description du plan :

4 Concurrence (3 pts)

Soient les transactions T_1 à T_5 accédant aux granules A à D. On note par $L_i(x)$ et $E_i(x)$ les opérations de lecture et d'écriture du granule x par la transaction T_i .

Question 15 (1½ points)

On considère l'exécution S_1 ci-dessous

$$L_3(D), E_5(D), E_2(B), L_4(C), L_5(D), E_1(A), E_4(C), L_2(C), E_5(A), E_2(C), L_1(B), L_4(A)$$

On considère que toutes les transactions demandent la validation juste après leur dernière opération.

Préciser pour chaque granule la séquence d'opération le concernant et les arcs de précédence induits par ces opérations, notés $T_i \rightarrow T_j$ en remplissant le tableau ci-dessous.

Réponse :

Granule	séquence d'opération	arcs de précédence
A		
B		
C		
D		

Répondre aux questions ci-dessous.

Réponse :

- Le graphe de précédence compte-t-il un circuit : oui / non. (Entourer la bonne réponse)

- Si oui, Dessinez-le

- Dire si l'exécution est sérialisable en justifiant votre réponse.

Question 16 (1½ points)On considère la séquence S_2 ci-dessous

$$L_2(A), L_4(C), L_1(B), L_3(B), E_4(C), V_4, E_2(A), E_3(B), L_1(A), V_1, E_3(C), V_3, E_2(B), V_2$$

On considère que le gestionnaire de concurrent utilise le protocole en 2 phases strict.

Réponse : Y a t il un interblocage : oui / non. (Entourer la bonne réponse)

En cas d'interblocage, donner le graphe des attentes.

Donner la séquence d'exécution en sortie. En cas d'interblocage, donner la séquence jusqu'à la dernière opération qui s'exécute.

5 Algèbre relationnelle (4 pts)

La plateforme de diffusion de musique *Teaser* gère des informations sur les albums de musique et les abonnés dans une base de données relationnelle avec 5 tables (les clés sont soulignées) :

Albums				
<u>idAlbum</u>	<u>idArtiste</u>	titreA	année	genre
B001Q	ar1	The 2nd Law	2012	Alternatif
B03EQ	ar1	The Resistance	2009	Rock
B002N	a2r	Ghost Stories	2014	Rock

Chansons			
<u>idC</u>	<u>idAlbum</u>	titreC	duréeC
t11	B001Q	Supremacy	04:55
t15	B001Q	Survival	04:17
t21	B03EQ	Uprising	05:30
t32	B002N	Magic	04:45
t37	B002N	Oceans	05:21

Artistes	
<u>idArtiste</u>	nom
ar1	Moose
a2r	Hotplay

Abonnes	
<u>idAbonne</u>	pseudo
ab1	francis234
ab2	céline145

Ecoutes			
<u>idAbonne</u>	<u>idC</u>	date	duréeE
ab1	t11	01/12/18 10:23	02:45
ab1	t21	01/12/18 10:27	05:20
ab2	t15	02/12/18 23:45	04:17
ab2	t21	02/12/18 23:45	05:30

La table **Albums** contient, pour chaque album, l'identifiant de l'album, l'identifiant de l'artiste correspondant et son année de parution. La table **Chansons** contient pour chaque chanson sur un album, l'identifiant unique de la chanson, l'identifiant de l'album, le titre de la chanson et sa durée en minutes. Les noms des artistes sont stockés dans la table **Artistes**.

Les pseudonymes des abonnés sont stockés dans la table **Abonnes** et la table **Ecoutes** contient pour chaque abonné les chansons qu'il a écoutées avec la date et la durée d'écoute. On suppose que le domaine des attributs **date** et **duree** est ordonnée et les valeurs peuvent être comparées avec les prédicats =, ≤, <, ≥ et >.

Exprimez toutes les requêtes suivantes en **algèbre relationnelle**. Vous pouvez décomposer chaque requête en plusieurs expressions en utilisant des variables.

Question 17 ($\frac{1}{2}$ point)

Les titres des chansons écoutées par francis234 avec les noms des artistes correspondants.

Réponse :

Question 18 ($\frac{1}{2}$ point)

Les noms des artistes écoutés par francis234 et céline145.

Réponse :

Question 19 (1 point)

Les titres des chansons qui n'ont jamais été écoutées jusqu'au bout par un abonné (les chansons qui n'ont jamais été écoutés sont également retournées).

Réponse :

Question 20 (1 point)

Les titres des chansons que francis234 a écoutées avant céline145.

Réponse :

Question 21 (1 point)

Les identifiants des abonnés qui ont écouté toutes les chansons de Hotplay.

Réponse :

6 Arbres B+ (3 pts)

On considère dans la suite des arbres B+ d'ordre 1 (chaque noeud peut avoir entre 1 et 2 valeurs).
Rappel : Pour une clé v_i contenue dans un noeud N , les clés contenues des sous-arbres à gauche de

v_i sont strictement inférieures à v_i , les clés des sous-arbres à droite de v_i sont supérieures ou égales à v_i .

On veut construire un arbre B+ pour indexer les valeurs entières suivantes : {14, 2, 18, 16, 10, 1, 20, 12}.

Rappel insertion : Lors d'un débordement d'une feuille F , on garde les 2 plus petites clés dans F , la valeur la plus grande sera stockée dans une nouvelle feuille.

Question 22 ($\frac{1}{2}$ point)

Dans quel ordre doit-on insérer les valeurs précédentes afin que les feuilles de l'arbre soient remplies au maximum (*i.e* chacune doit contenir 2 valeurs)? Dessinez l'arbre A_0 obtenu (l'arbre doit contenir uniquement les valeurs précédentes).

Réponse :

Ordre d'insertion : . . .

Arbre B+ :

Question 23 (1 point)

On considère l'arbre A_1 suivant dans lequel on insère les valeurs suivantes **en respectant l'ordre d'insertion** : {20, 18, 16, 14}. Quel est l'arbre A_2 qui sera obtenu après l'insertion de ces valeurs ?

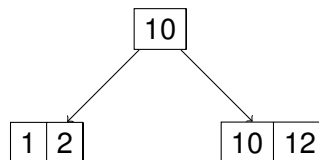


FIGURE 1 – Arbre A_1

Réponse : *Arbre A_2* :

Question 24 ($\frac{1}{2}$ point)

Donnez un ordre d'insertion, **autre que l'ordre croissant**, des valeurs $\{20, 18, 16, 14\}$ dans A_1 qui produit comme résultat A_0 .

Réponse : *Ordre d'insertion : ...*

Question 25 (1 point)

On considère l'arbre A_3 suivant dans lequel on supprime la clé 18. Dessinez l'arbre A_4 obtenu après cette suppression. On considère tout d'abord la redistribution avec le frère à gauche, ensuite avec celui de droite. Si cela n'est pas possible on considère la fusion avec le frère de gauche et ensuite avec celui de droite.

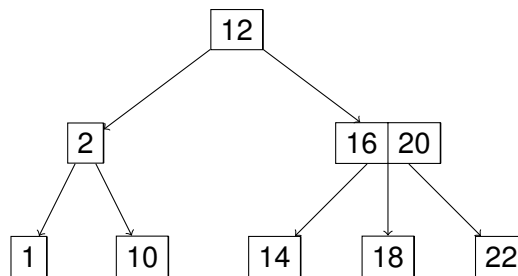


FIGURE 2 – Arbre A_3

Réponse :

Arbre A_4 :