Numéro d'anonymat:

Université Bordeaux 1 — Master d'informatique — UE Bases de Données
Sujet et correction de l'examen du 27 mai 2004 8h00–9h30 (sans documents)

Sauf mention contraire en caractères **gras**, *la place sur ce sujet suffit pour répondre aux questions posées. Si une question est ambiguë, on précisera comment on l'a comprise.*

Conception de schémas relationnels

Rappel: Soit I un ensemble fini d'attributs, soit J_k des sous-ensembles de I pour $1 \leq k \leq K$. Soit R un schéma relationnel sur I. La décomposition de $R(I)$ en $(R_k)_{1 \leq k \leq K}$ est dite *sans perte par jointure* si et seulement si $\bowtie_{k=1}^{k=K} \pi_{J_k}(R) = R$.

Exercice A

On considère le schéma relationnel suivant, Cours, Prof, Heure, Salle, Etudiant, Note dont les dépendances fonctionnelles sont

Cours	$\rightarrow$	Prof	Heure Salle	$\rightarrow$	Cours
Heure Prof	$\rightarrow$	Salle	Cours Etudiant	$\rightarrow$	Note
Heure Etudiant	$\rightarrow$	Salle	Heure Salle Etudiant	$\rightarrow$	Note

(A.i) Donner les clefs minimales de cette relation.

> Une unique clef minimale: HE

(A.ii) Pourquoi ce schéma n'est-il pas en 3NF?

> À cause de la dépendance fonctionnelle $C \rightarrow P$:
> C ne contient pas de clef, et P n'appartient pas à une clef.

(A.iii) Donner ici une décomposition en 3NF. **(on précisera sur sa copie comment elle est obtenue)**

> Il faut trouver une couverture minimale de l'ensemble de dépendances fonctionnelles. On peut supprimer $HSE \rightarrow N$, puisque HE est une clef. Ensuite, on peut voir que les dépendances fonctionnelles restantes constituent une couverture minimale. On peut appliquer l'algorithme vu en cours et en TD pour s'assurer que les autres dépendances constituent une couverture minimale :
>
> – On essaie de supprimer des dépendances:
>   – Si l'on supprimait la dépendance $C \rightarrow P$ alors l'attribut P ne serait plus déterminé par aucun ensemble d'attributs ne le contenant pas: il ne serait plus dans aucun membre droit. Le même raisonnement montre qu'on ne peut pas non plus supprimer $CE \rightarrow N$, pas plus que $HS \rightarrow C$.
>   – Si on supprimait $HP \rightarrow S$, on en retrouverait pas cette dépendance car la clôture de HP par les autres dépendances est HP.
>   – Si on supprimait $HE \rightarrow S$, on en retrouverait pas cette dépendance car la clôture de HE par les autres dépendances est HE.
>
> – On essaie de supprimer des attributs dans les membres de gauche, mais dès que l'on en supprime un, on obtient une dépendance qui n'est pas valide — c.-à-d.qui n'est pas conséquence des dépendances de départ.
>
> – On ne peut donc ni supprimer de dépendance, ni d'attribut dans un membre de gauche: on a une couverture minimale.
>
> On décompose le schéma relationnel avec une relation par dépendance fonctionnelle: CP,HSC,HPS,CEN,HES.

(A.iv) La décomposition préserve-t-elle les dépendances fonctionnelles?

> Par construction, les dépendances fonctionnelles sont préservées — en projetant sur chaque schéma on obtient la dépendance fonctionnelle qui a servi à le définir et donc toutes les dépendances fonctionnelles.

(A.v) Y a-t-il perte par jointure? Sinon, modifier le schéma pour qu'il n'y ait pas de perte par jointure.

> A priori, pour être sûr qu'il n'y a pas de perte par jointure, il faut ajouter une relation correspondant à une clef. Ici, ce n'est pas la peine, puisque la table HES contient la clef HE. On peut aussi appliquer l'algo matriciel, cf. ci-après.

Exercice B

On décompose le schéma relationnel $R(A,B,C,D)$ muni des dépendances fonctionnelles

A → B   B → C   C → A

(B.i) Montrer, avec les axiomes d'Armstrong, que la dépendance $C \to B$ est conséquence de ces dépendances.

> De $C \to A$ et $A \to B$ on obtient, par transitivité $C \to B$.

(B.ii) Si on décompose ce schéma en $S(A,B)$ et $T(B,C,D)$ la dépendance fonctionnelle $C \to A$ est-elle préservée? Pourquoi? **(Si l'on manque de place, on utilisera sa copie)**

> Oui. En effet, c'est la clôture de l'ensemble des dépendances fonctionnelles que l'on projette sur AB et BCD. La clôture de l'ensemble de dépendances fonctionnelles
>
> $$F = \{A \to B, B \to C, C \to A\}$$
>
> est
>
> $$Cl(F) = \{A \to B, B \to C, C \to A, C \to B, B \to A, A \to C\}$$
>
> La projection de $Cl(F)$ sur AB est $\pi_{AB}Cl(F) = \{A \to B, B \to A\}$, tandis que la projection de $Cl(F)$ sur BCD est
>
> $$\pi_{BCD}Cl(F) = \{B \to C, C \to B\}$$
>
> On a donc $C \to B$ dans $\pi_{BCD}Cl(F)$ et $B \to A$ dans $\pi_{AB}Cl(F)$, d'où, par transitivité $C \to A$.

(B.iii) Si on décompose ce schéma en $S(A,B)$ et $T(B,C,D)$ y a-t-il perte par jointure?

L'algorithme vu en cours donne la suite de tableaux suivants.
Initialisation:

	A	B	C	D
S	a_1	a_2	$b_{1,3}$	$b_{1,4}$
T	$b_{2,1}$	a_2	a_3	a_4

Comme on a la dépendance fonctionnelle $B \to A$ on obtient:

	A	B	C	D
S	a_1	a_2	$b_{1,3}$	$b_{1,4}$
T	a_1	a_2	a_3	a_4

et comme on a aussi la dépendance fonctionnelle $B \to C$ on obtient :

	A	B	C	D
S	a_1	a_2	a_3	$b_{1,4}$
T	a_1	a_2	a_3	a_4

Ce dernier tableau définit une relation qui satisfait toutes les dépendances fonctionnelles. Comme une ligne est a_1,a_4,a_3,a_4 (ce dont on peut se rendre compte dès le pas précédent), il n'y a pas de perte par jointure.
On peut aussi utiliser le théorème de Heath: comme $AB \cap BCD = B$ et $B \to A = AB - BCD$ on a

$$S \cap T \to S - T$$

Le théorème de Heath permet d'affirmer que la decomposition de $R(A,B,C,D)$ en $S(A,B)$ et $T(B,C,D)$ est sans perte par jointure.

SQL et algèbre relationnelle

Exercice C

On considère les tables suivantes:

R :

A	B
a_1	b_2
a_2	b_1
a_4	b_1

S :

A	B
a_2	b_1
a_4	b_3
a_5	b_1

U :

A	B
a_1	b_2
a_2	b_3
a_4	b_2
a_5	b_3

(C.i) Que valent

1. $T = (\sigma_{B=b_1} U)$

$T = \emptyset$

2. $(R \cap S) \cup (R \cap T)$

$(R \cap S) \cup (R \cap T) = \{(a_2, b_1)\}$

(C.ii) Quelle est la réponse à la requête SQL suivante (on justifiera brièvement sa réponse):
$SELECT \quad R.A, \quad R.B$
$FROM \quad R, \quad S, \quad T \; AS \, (SELECT \quad * \quad FROM \quad U \quad WHERE \quad B = b_1)$
$WHERE \quad (R.A = S.A \quad AND \quad R.B = S.B) \quad OR \quad (R.A = T.A \quad AND \quad R.B = T.B)$

La réponse est $\emptyset$. En effet SQL commence par calculer le produit cartésien $R \times S \times T$ dans lequel il sélectionne les n-uplets satisfaisant la condition, avant de projeter sur les attributs demandés. Comme $T = \emptyset$ ce produit est vide et, par conséquent, le résultat de la requête est aussi vide.

Datalog

Exercice D

Un jeu se déroule sur un quart de plan quadrillé $\{(i,j) \in N \times N | i,j \geq O\}$ en plaçant des arêtes blanches (B) ou noires (N) entre les intersections du quadrillage.

On décrit l'état du damier à un moment de la partie par une table $P(i,j,i',j',C)$ qui indique qu'il y a une arête de couleur C (C vaut B ou C) entre l'intersection i,j et l'intersection i',j' ce qui n'est possible que si les deux intersections sont voisines en suivant le quadrillage, c'est-à-dire

- $i = i'$ et $|j - j'| = 1$
- $j = j'$ et $|i - i'| = 1$

[En d'autres termes, on ne peut placer d'arêtes sur les diagonales.]

On dit qu'il existe un chemin de l'intersection $P_0 = (x_0,y_0)$ à l'intersection $P_n = (x_n,y_n)$ s'il existe une suite d'intersections $P_1 = (x_1,y_1),...,P_{n-1} = (x_{n-1},y_{n-1})$ tel que pour tout k compris entre 0 et $n-1$ il existe une arête de couleur B ou N entre $P_k = (x_k,y_k)$ et $P_{k+1} = (x_{k+1},y_{k+1})$. La longueur d'un tel chemin est n.

On dit que le chemin est de couleur B s'il ne comporte que des arêtes de couleur B, et de couleur N s'il ne comporte que des arêtes de couleur N.

En prenant soin d'écrire des clauses ou ensembles de clauses correctes (en particulier lorsqu'on utilise simultanément récursion et négation), formuler en datalog les requêtes suivantes:

(D.i) Quels sont les couples d'intersections entre lesquels il y a une arête B?

```
B1(x,y,u,v):-P(x,y,u,v,'B').
```

(D.ii) Quels sont les couples d'intersections reliées par un chemin B de longueur inférieure ou égale à 2?

```
Binfeg2(x,y,u,v):-B1(x,y,u,v).
Binfeg2(x,y,u,v):-B1(x,y,a,b) AND B1(a,b,u,v).
```

(D.iii) Quels sont les couples d'intersections reliées par un chemin B de longueur exactement 3 et qui sont situés sur la même ligne horizontale?

```
B3H(x,y,x,v):-B1(x,y,a,b) AND B1(a,b,c,d) AND B1(c,d,x,v).
```

(D.iv) Quels sont les couples d'intersections reliées par un chemin B?

```
B(x,y,u,v):-B1(x,y,u,v).
B(x,y,u,v):-B1(x,y,a,b) AND B(a,b,u,v).
```

(D.v) Quels sont les couples d’intersections reliées par un chemin utilisant indifféremment des arêtes B ou N?

```
BouN(x,y,u,v):-P(x,y,u,v,_).
BouN(x,y,u,v):-P(x,y,a,b,_) AND BouN(a,b,u,v,_).
```

(D.vi) Quels sont les couples d’intersections qui ne sont reliées que par des chemins B?

On définit `Nsupeg1`, un prédicat à quatre arguments, qui décrit les couples d’intersections qui sont reliées par des chemins contenant au moins une arête noire — notez qu’il faut deux cas dans la récurrence: le sous chemin contenant au moins une arête noire peut être au début ou à la fin.

```
Nsupeg1(x,y,u,v):-P(x,y,u,v,'N').
Nsupeg1(x,y,u,v):-P(x,y,a,b,_) AND Nsupeg1(a,b,u,v).
Nsupeg1(x,y,u,v):-Nsupeg1(x,y,a,b) AND P(a,b,u,v,_).
```

La solution, le prédicat `queB` est l’ensemble des couples d’intersections reliées par un chemin blanc qui ne sont pas reliées par un chemin contenant au moins une arête noire, soit:

```
queB(x,y,u,v):-B(x,y,u,v) AND NOT Nsupeg1(x,y,u,v).
```

On peut vérifier qu’il n’y a pas de conflit entre récursion et négation, puisque le graphe des dépendances ne contient aucun cycle orienté utilisant une flêche négative (avec le signe « $\ominus$ »): les seuls cycles orientés sont les boucles, lesquelles n’ont pas de « $\ominus$ ».

$$\begin{array}{ccccc} & & P & & \\ & \nearrow & \uparrow & \nwarrow & \\ B & \leftarrow & queB & {}_{\ominus}\!\rightarrow & Nsupeg1 \\ \circlearrowleft & & & & \circlearrowleft \end{array}$$

Optimisation de requêtes

Exercice E

On dispose de deux relations R(A,B,C) et S(C,D,E).

Pour chacune des requêtes suivantes, donner une requête équivalente plus efficace.

(E.i) $\pi_A(R \bowtie S)$

$\pi_A(\pi_{AC}(R) \bowtie \pi_C(S))$

(E.ii) $\pi_{ABE}(R \bowtie S)$.

$\pi_{ABE}(R \bowtie \pi_{CE}(S))$

(E.iii) $\sigma_{A=a}(R \bowtie S)$

$\sigma_{A=a}(R) \bowtie S$

(E.iv) $\sigma_{[C=c]}(R \bowtie S)$

$\sigma_{[C=c]}(R) \bowtie \sigma_{[C=c]}(S)$

(E.v) $\sigma_{[C=c\ \ AND\ \ E=e]}\pi_{A,C,E}(R \bowtie S)$

$\pi_{AC}\sigma_{[C=c]}(R) \bowtie \pi_{CE}\sigma_{[C=c\ AND\ E=e]}(S)$
Remarque $\pi_{CE}\sigma_{[C=c\ AND\ E=e]}(S)$ vaut $\{(c,e)\}$ou $\emptyset$.

Ensembles et multi-ensembles

Exercice F

Dans chacun des deux cas suivants, dire si les expressions $A = ((R \cup S) - T)$ et $B = ((R - T) \cup (S - T))$ sont égales — on donnera une rapide justification de l'égalité ou des valeurs aux relations R,S,T qui conduisent à des valeurs différentes pour les expressions A et B.

(F.i) Lorsque R,S,T sont des ensembles de n-uplets et « $\cup$ » et « $-$ » l'union et la différence entre ensembles

Lorsque R, S et T sont des ensembles, on a $A = B$. Un simple diagramme de Venn suffit, mais sur un ordinateur il est plus simple de taper du texte:
$\boxed{A \subset B}$ Soit $x \in A$ alors $x \in R \cup S$ (1) et $x \notin T$ (2). D'après (1) on a au moins l'une des deux propriétés suivantes $x \in R$ (3), $x \in S$ (4). Si (3) alors $x \in R - T$ en utilisant (2) et si (4) alors $x \in S - T$ an utilisant (2). Dans un cas comme dans l'autre, $x \in B$.
$\boxed{B \subset A}$ Si $x \in B$ alors on a au moins l'une des propriétés suivantes: $x \in R - T$ (5), $x \in S - T$ (6). Si (5) on a $x \in R$ et $x \notin T$ et donc $x \in (R \cup S)$ and $x \notin T$, c'est-à-dire $x \in A$. Si (6), on a $x \in S$ et $x \notin T$ et donc $x \in (R \cup S)$ and $x \notin T$, c'est-à-dire $x \in A$.Dans les deux cas, on a $x \in A$.

(F.ii) Lorsque R,S,T sont des mutli-ensembles de n-uplets et « $\cup$ » et « $-$ » l'union et la différence entre multi-ensembles.

Si R,S,T sont des multiensembles, les expressions A et B ne sont pas nécessairement égales.
Soit $R = S = T = \{(a,b)\}$.
$B = \emptyset$ puisque $R - T = S - T = \emptyset$.
Par contre $A = \{(a,b)\}$ puisque $R \cup S = \{(a,b),(a,b)\}$.

S'il reste du temps après avoir traité ces exercices...

On décomposera en BCNF le schéma relationnel donné dans le premier exercice et on vérifiera la préservation des dépendances — **sur sa copie**.

Comme dit dans la question cet exercice ne concernait que ceux qui auraient eu tout fait — d'autres s'y sont quand même essayé et ont obtenu quelques points. Il était en tout cas hors de question de finir cet exercice très long. Un corrigé avec quelques explications est rédigé à part et est aussi accessible en ligne.