

Bases de Données Avancées : TD1 - Corrigé

Exercice 1.1

1. $q(x) \leftarrow \text{Films}('Brazil', x, y)$.
2. $q(x) \leftarrow \text{Guide}(x, 'Brazil', y)$.
3. $q(x_1, x_2) \leftarrow \text{Lieux}('CinemaBordelais', x_1, x_2)$.
4. $q(x_1, x_2) \leftarrow \text{Lieux}(x_1, x_2, y_1), \text{Guide}(x_1, y_2, y_3), \text{Films}(y_2, 'TerryGilliam', y_4)$.
5. $q \leftarrow \text{Lieux}(y_1, 'Bordeaux', y_2), \text{Guide}(y_1, y_3, y_4), \text{Films}(y_3, 'TerryGilliam', y_5)$.
6. $q(x_1, x_2) \leftarrow \text{Films}(y_1, x_1, x_2), \text{Films}(y_2, x_2, x_1)$.
7. $q(x) \leftarrow \text{Films}(y, x, x)$.
8. $q(x_1, x_2) \leftarrow \text{Films}(y_1, y_2, x_1), \text{Films}(y_1, y_3, x_2)$.
9. $q('ApocalypseNow', 'Coppola') \leftarrow$.
10. Non exprimable en calcul conjonctif (il existe des instances sans la constante 'Brazil' qui satisfont la requête). En calcul relationnel :
 $\{x \mid \exists y_1, y_2 \text{ Guide}(x, 'Brazil', y_1) \vee \text{Guide}(x, 'ApocalypseNow', y_2)\}$
11. $q(x_1, x_2) \leftarrow \text{Films}('ApocalypseNow', x_2, x_1)$.
12. Non exprimable en calcul conjonctif car non monotone. Par exemple si on prend les instances :
 I telle que $\text{Films} = \{ 'IndianaJones' 'Spielberg' 'SeanConnery' \}$
 I' telle que $\text{Films} = \{ 'IndianaJones' 'Spielberg' 'SeanConnery' \}$
 $\{ 'IndianaJones' 'Spielberg' 'HarrisonFord' \}$
 On a bien $I \subseteq I'$ mais pas $q(I) \subseteq q(I')$ la requête n'est donc pas monotone et donc par conséquent pas conjonctive. En calcul relationnel :
 $\{x \mid \exists y \text{ Films}(x, 'Spielberg', y) \wedge \neg \text{Films}(x, 'Spielberg', 'HarrisonFord')\}$
13. Non exprimable en calcul conjonctif car non monotone. En calcul relationnel :
 $\{x \mid \exists y \text{ Films}(x, 'Gilliam', y) \vee \neg \text{Films}(x, y, 'HarrisonFord')\}$
14. Non exprimable en calcul relationnel.
15. Non exprimable en calcul conjonctif car non monotone. En calcul relationnel :
 $\{x \mid \exists y_1, y_2 (\text{Guide}(y_1, x, y_2) \wedge \neg \exists y_3, y_4 (y_3 \neq y_1 \wedge \text{Guide}(y_3, x, y_4)))\}$

Exercice 1.2

1. On sait que $\bar{a} \in q(\mathbf{I})$ ssi il existe un homomorphisme de \mathbf{I}_q dans \mathbf{I} tel que $h(\bar{x}) = \bar{a}$. Il en découle que toutes les constantes de \bar{a} sont dans $\text{adom}(\mathbf{I})$. Cet ensemble étant par définition fini, il n'existe qu'un nombre fini de possibilités pour \bar{a} .

2. Soit n l'arité de q , $|q(\mathbf{I})| \leq (|adom(\mathbf{I})|^n)$.

Exercice 1.3

1. CF exercice précédent.
2. Il suffit de considérer les constantes présentes dans $adom(q)$ et $adom(\mathbf{I})$.

Exercice 1.4

1. Considérons $q(x) \rightarrow x = x$. Pour tout \mathbf{I} on a $q(\mathbf{I}) = \mathbf{dom}$ qui est infini par définition.
2. Toute variable z utilisée dans la requête doit soit :
 - être utilisée dans une expression $z = 'c'$ où $'c'$ est une constante.
 - ou être utilisée dans une expression $R(\dots, z, \dots)$ où R est un nom de relation.

Exercice 1.5

1. La requête $q \rightarrow R(x) \wedge R(y) \wedge x \neq y$ n'est pas exprimable sans l'inégalité.
2. On note $C_=, C_{=,\neq}, C_{=,<}, C_{<}$ et $C_{=,\neq,<}$ les différentes variantes. On a :

$$\begin{array}{lcl}
 C_= & \subsetneq & C_{=,\neq} \\
 C_= & \subsetneq & C_{=,<} \\
 C_{=,<} & = & C_{<} \\
 C_{=,\neq} & \subsetneq & C_{=,\neq,<} \\
 C_{=,<} & \subsetneq & C_{=,\neq,<}
 \end{array}$$

Toutes les autres inclusions sont fausses.

Exercice 1.6

– $q_2 \subseteq q_3$:

$$\begin{array}{l}
 x_1 \rightarrow x_1 \\
 x_2 \rightarrow x_2 \\
 y_1 \rightarrow y_2 \\
 y_2 \rightarrow y_2 \\
 y_3 \rightarrow y_3 \\
 y_4 \rightarrow y_3
 \end{array}$$

– $q_3 \subseteq q_2$:

$$\begin{array}{l}
 x_1 \rightarrow x_1 \\
 x_2 \rightarrow x_2 \\
 y_2 \rightarrow y_1 \\
 y_3 \rightarrow y_3
 \end{array}$$

– $q_4 \subseteq q_2$

$$\begin{array}{l}
 x_1 \rightarrow x_1 \\
 x_2 \rightarrow x_2 \\
 y_2 \rightarrow y_1 \\
 y_3 \rightarrow 'cst'
 \end{array}$$

Exercice 1.7

Tester l'équivalence entre q, q' revient à tester l'inclusion dans les deux sens. Le problème d'inclusion étant dans NP, l'équivalence l'est donc aussi. On prouve maintenant que l'équivalence est NP-difficile en réduisant le problème d'inclusion (dont on sait qu'il est NP-difficile) au problème d'équivalence. Soit q, q' , on considère les requête $r = q$ et $r' = q \wedge q'$. Il n'est pas difficile de vérifier que $q \subseteq q'$ ssi $r \sim r'$. De plus calculer r, r' à partir de q, q' se fait en temps polynomial. On a donc bien une réduction PTIME de l'inclusion à l'équivalence. On en conclut que l'équivalence entre requêtes est un problème NP-complet.