

14/2/2017 Úvod do databáz, skúškový test, max **60** bodov

1. Uvažujte databázu bez duplikátov a null hodnôt: `capuje(Krcma, Alkohol)`,  
`lubi(Pijan, Alkohol)`, `navstivil(Idn, Pijan, Krcma)`, `vypil(Idn, Alkohol, Mnozstvo)`.

Platí: `Idn → Pijan, Krcma`; `Idn, Alkohol → Mnozstvo`; `Mnozstvo > 0`.

a) Sformulujte v Datalogu (**6**) bezpečný dotaz na dvojice `[P, K]` také, že pijan `P` dohromady vypil počas niektorých dvoch návštev v krčme `K` všetky svoje obľúbené alkoholy, pričom toto sa pijanovi `P` nepodarilo počas žiadnej (jednej) návštevy krčmy `K`.

```
answer(P, K) ←  
  navstivil(I1, P, K),  
  navstivil(I2, P, K),  
  not niekory_oblubeny_nevypil(P, I1, I2),  
  not raz_vsetky_oblubene_vypil(P, K).
```

```
niekory_oblubeny_nevypil(P, I1, I2) ←  
  lubi(P, A),  
  navstivil(I1, P, _),  
  navstivil(I2, P, _),  
  not v(I1, A),  
  not v(I2, A).
```

```
v(I, A) ←  
  vypil(I, A, _).
```

```
raz_vsetky_oblubene_vypil(P, K) ←  
  navstivil(I, P, K),  
  niekory_oblubeny_nevypil2(P, I).
```

```
niekory_oblubeny_nevypil2(P, I) ←  
  navstivil(I, P, _),  
  lubi(P, A),  
  not v(I, A).
```

b) Sformulujte v relačnom kalkule (6), SQL (6) a relačnej algebre (6) dotaz na krčmy, v ktorých niektorý pijan vypil rum v celkovom množstve väčšom ako 142.

Relačný kalkul:

{K:

$\exists P \exists S$

$\heartsuit I, S = \text{sum}(M) (\text{navstivil}(I, P, K) \wedge \text{vypil}(I, \text{rum}, M))$

$\wedge S > 142$

}

SQL:

```
select n.Krcma
```

```
from navstivil n, vypil v
```

```
where n.Idn = v.Idn and v.Alkohol = 'rum'
```

```
group by n.Krcma, n.Pijan
```

```
having sum(v.Mnozstvo) > 142
```

Relačná algebra:

$\Pi_{Krcma} (\sigma_{S > 142} (\Gamma_{Pijan, Krcma, S = \text{sum}(Mnozstvo)} (\text{navstivil} \bowtie \sigma_{Alkohol = 'rum'} (\text{vypil}))))$

2. Daná je relácia  $r(A, B, C, D, E, F, G, H)$  s funkčnými závislosťami  
 $A \rightarrow G, AD \rightarrow B, AC \rightarrow BDE, E \rightarrow B, BG \rightarrow CF, BE \rightarrow D, BH \rightarrow E, BCD \rightarrow GH, BDG \rightarrow A.$

a) Dekomponujte  $r$  do tretej normálnej formy, bezstratovo a so zachovaním všetkých funkčných závislostí. (6)

Jedno z minimálnych pokrytí danej množiny funkčných závislostí:

$\{A \rightarrow G, AD \rightarrow B, AC \rightarrow E, E \rightarrow B, BG \rightarrow C, BG \rightarrow F, E \rightarrow D, BH \rightarrow E, BCD \rightarrow G, BCD \rightarrow H, BDG \rightarrow A\}$

**3NF dekompozícia podľa tohto minimálneho pokrytia (ABDG je nadkľúč v  $r$ ):**

**ABDG, ACE, BCDGH, BCFG, BEH, DE**

b) Dekomponujte  $r$  do Boyce-Coddovej normálnej formy, bezstratovo. Snažte sa vyhnúť zlomeniu funkčných závislostí. (6)

Začnime s 3NF dekompozíciou z úlohy a).

ABDG nie je v BCNF, lebo platí  $A \rightarrow G$ , pričom  $A$  nie je nadkľúč v ABDG. Dekomponujeme do ABD, AG.

ABD je v BCNF.

AG je v BCNF.

ACE je v BCNF.

BCDGH nie je v BCNF, lebo platí  $BG \rightarrow C$ , pričom  $BG$  nie je nadkľúč v BCDGH. Dekomponujeme do BCG, BDGH.

BCG je v BCNF.

BDGH nie je v BCNF, lebo platí  $BH \rightarrow D$ , pričom  $BH$  nie je nadkľúč v BDGH. Dekomponujeme do BDH, BGH.

BDH je v BCNF.

BGH je v BCNF.

BCFG je v BCNF.

BEH nie je v BCNF, lebo platí  $E \rightarrow B$ , pričom  $E$  nie je nadkľúč v BEH. Dekomponujeme do BE, EH.

BE je v BCNF.

EH je v BCNF.

DE je v BCNF.

**BCNF dekompozícia:**

**ABD, AG, ACE, BCFG, BDH, BE, BGH, DE, EH**

3. Daný je Datalogový program nad extenzionálnou databázou  $\{e(., .)\}$ :

$p(A, B) \leftarrow e(A, B)$ .

$p(C, D) \leftarrow p(C, E), p(E, D)$ .

$q(X, Y, Z) \leftarrow p(X, Z), p(Z, Z), p(Z, Y)$ .

a) Vypočítajte výsledok dotazu „? q(X, Y, 2)“ pre reláciu

$e(., .) = \{[0, 1], [1, 4], [2, 14], [4, 16], [14, 2], [16, 14]\}$ .

Okomentujte priebeh výpočtu. (6)

Program budeme počítat' semi-naivnou evaluáciou. Keďže relácia priradená predikátu  $p(., .)$  závisí len od obsahu relácií  $p(., .)$  a  $e(., .)$  (pričom  $e$  sa nemení), vypočítame najskôr obsah relácie  $p$ .

Inicializácia:

$p(P1, P2) = \emptyset$ ;

$q(Q1, Q2) = \emptyset$ ;

/\* Teraz budeme iterovať  $p(P1, P2) = P_{(P1, P2)}(e) \cup \Pi_{p1.P1, p2.P2} (P_{p1}(p) \bowtie_{p1.P2 = p2.P1} P_{p2}(p))$ ;  
až kým sa  $p(P1, P2)$  prestane meniť. \*/

Po 1. iterácii:

$p(P1, P2) = \{[0, 1], [1, 4], [2, 14], [4, 16], [14, 2], [16, 14]\}$ .

Po 2. iterácii:

$p(P1, P2) = \{[0, 1], [0, 4], [1, 4], [1, 16], [2, 2], [2, 14], [4, 14], [4, 16], [14, 2], [14, 14], [16, 2], [16, 14]\}$ .

Po 3. iterácii:

$p(P1, P2) = \{[0, 1], [0, 4], [0, 14], [0, 16], [1, 2], [1, 4], [1, 14], [1, 16], [2, 2], [2, 14], [4, 2], [4, 14], [4, 16], [14, 2], [14, 14], [16, 2], [16, 14]\}$ .

Po 4. iterácii:

$p(P1, P2) = \{[0, 1], [0, 2], [0, 4], [0, 14], [0, 16], [1, 2], [1, 4], [1, 14], [1, 16], [2, 2], [2, 14], [4, 2], [4, 14], [4, 16], [14, 2], [14, 14], [16, 2], [16, 14]\}$ .

Po 5. iterácii:

$p(P1, P2) = \{[0, 1], [0, 2], [0, 4], [0, 14], [0, 16], [1, 2], [1, 4], [1, 14], [1, 16], [2, 2], [2, 14], [4, 2], [4, 14], [4, 16], [14, 2], [14, 14], [16, 2], [16, 14]\}$ .

/\* Vypočítame  $q(Q1, Q2, Q3) = \Pi_{p1.P1, p2.P2} ($   
 $(P_{p1}(p) \bowtie_{p1.P2 = p2.P1 \wedge p2.P1 = p2.P2} P_{p2}(p)) \bowtie_{p2.P2 = p3.P1} P_{p3}(p))$   
); \*/

$q(Q1, Q2, Q3) = \{[0, 2, 2], [0, 2, 14], [0, 14, 2], [0, 14, 14],$   
 $[1, 2, 2], [1, 2, 14], [1, 14, 2], [1, 14, 14],$   
 $[2, 2, 2], [2, 2, 14], [2, 14, 2], [2, 14, 14],$   
 $[4, 2, 2], [4, 2, 14], [4, 14, 2], [4, 14, 14],$   
 $[14, 2, 2], [14, 2, 14], [14, 14, 2], [14, 14, 14],$   
 $[16, 2, 2], [16, 2, 14], [16, 14, 2], [16, 14, 14]\}$

/\* Výsledkom dotazu je  $\Pi_{Q1, Q2} (\sigma_{Q3=2} (q(Q1, Q2, Q3)))$  \*/

Výsledkom dotazu „? q(X, Y, 2)“ pre danú reláciu je

$\{[0, 2], [0, 14], [1, 2], [1, 14], [2, 2], [2, 14], [4, 2], [4, 14], [14, 2], [14, 14], [16, 2], [16, 14]\}$ .

b) Nahraďte prvé pravidlo programu pravidlom

$p(A, B) \leftarrow e(A, B), q(\_, \_, \_)$ .

Riešte úlohu a) pre takto modifikovaný program. (6)

Seminaivná evaluácia začína s prázdnyimi reláciami p a q. Join prázdnej relácie s ľubovoľnou inou je prázdna relácia.

**Výsledkom dotazu je prázdna relácia (dokonca pre ľubovoľné naplnenie e(.,.)).**

4. Databázový systém spadol. Pred jeho opätovným spustením obsahuje log-file na disku nasledujúce záznamy: [T1, start], [T1, X, 13, 27], [T1, Y, 14, 4], [T2, start], [T2, Z, 19, 9], [T3, start], [T2, W, 3, 6], [T1, commit], [T3, X, 27, 16], [T3, commit].

Log-file rastie zľava doprava. Záznamy týkajúce sa operácií write obsahujú na treťom mieste starú hodnotu, na štvrtom mieste novú hodnotu.

a) Uved'te všetky hodnoty, ktoré môžu byť zapísané na disku v dátových objektoch X, Y, Z, W (individuálne pre X, Y, Z, W) počas trvania výpadku. Vysvetlite. (6)

Cache manager vo všeobecnosti nemusí, ale smie, uložiť pri zápise do dátového objektu uložiť novú hodnotu na disk (nová hodnota môže byť zapamätaná len v RAM). Ak nie sú na správanie cache managera kladené špeciálne predpoklady, tak scheduler aktualizáciu úmysel najskôr zapíše do logu a až potom nechá cache managera aktualizáciu urobiť (write-ahead logging).

**X** môže mať na disku nasledujúce hodnoty: **13** (pôvodná hodnota), **27** (hodnota zapísaná T1), **16** (hodnota zapísaná T3).

**Y** môže mať na disku nasledujúce hodnoty: **14** (pôvodná hodnota), **4** (hodnota zapísaná T1).

**Z** môže mať na disku nasledujúce hodnoty: **19** (pôvodná hodnota), **9** (hodnota zapísaná T2).

**W** môže mať na disku nasledujúce hodnoty: **3** (pôvodná hodnota), **6** (hodnota zapísaná T2).

b) Uved'te a okomentujte akcie, ktoré systém urobí počas obnovy pri čítaní jednotlivých záznamov z daného log-file. (6)

Zostupný (UNDO) prechod cez log-file (vpravo je akcia):

	undo_list = $\emptyset$ ;
	redo_list = $\emptyset$ ;
[T3, commit]	redo_list = {T3};
[T3, X, 27, 16]	
[T1, commit]	redo_list = {T1, T3};
[T2, W, 3, 6]	undo_list = {T2}; W = 3;
[T3, start]	
[T2, Z, 19, 9]	Z = 19;
[T2, start]	undo_list = $\emptyset$ ;
[T1, Y, 14, 4]	
[T1, X, 13, 27]	
[T1, start]	
<BOF>	začni vzostupný (REDO) prechod
[T1, start]	
[T1, X, 13, 27]	X = 27;
[T1, Y, 14, 4]	Y = 4;
[T2, start]	
[T2, Z, 19, 9]	
[T3, start]	
[T2, W, 3, 6]	
[T1, commit]	redo_list = {T3};
[T3, X, 27, 16]	X = 16;
[T3, commit]	redo_list = $\emptyset$ ;

Vyššieuvedené priradenia (napr. W = 3) znamenajú zápisy do príslušných dátových objektov. Opäť, cache manager rozhoduje o tom, či tieto hodnoty uloží (aj) na disk. Vo všeobecnosti môže, ale nemusí.