

14/1/2021 Úvod do databáz, skúškový test, max 60 bodov

1. Uvažujte databázu bez duplikátov a null hodnôt:  $\text{capuje}(\text{Krcma}, \text{Alkohol})$ ,  $\text{lubi}(\text{Pijan}, \text{Alkohol})$ ,  $\text{navstivil}(\text{Idn}, \text{Pijan}, \text{Krcma})$ ,  $\text{vypil}(\text{Idn}, \text{Alkohol}, \text{Mnozstvo})$ .

Platí:  $\text{Idn} \rightarrow \text{Pijan}, \text{Krcma}$ ;  $\text{Idn}, \text{Alkohol} \rightarrow \text{Mnozstvo}$ ;  $\text{Mnozstvo} > 0$ .

a) Sformulujte bezpečný dotaz v Datalogu (6) a relačnej algebre (6) na dvojice pijanov [P1, P2] také, že pijani P1 a P2 navštívili tie isté krčmy (rovnakú neprázdnu množinu krčiem); a zároveň každý z nich vypil, nezávisle na tom druhom pijanovi, len také alkoholy, ktoré ľúbi (čo platí aj pre totálnych abstinentov).

Datalog:

$\text{answer}(\text{P1}, \text{P2}) :-$

$\text{n}(\text{P1}, \_)$ ,  
 $\text{n}(\text{P2}, \_)$ ,  
 $\text{not navstivil\_nenavstivil}(\text{P1}, \text{P2})$ ,  
 $\text{not navstivil\_nenavstivil}(\text{P2}, \text{P1})$ ,  
 $\text{not vypil\_nelubi}(\text{P1})$ ,  
 $\text{not vypil\_nelubi}(\text{P2})$ .

$\text{navstivil\_nenavstivil}(\text{P1}, \text{P2}) :-$

$\text{n}(\text{P1}, \text{K})$ ,  
 $\text{n}(\text{P2}, \_)$ ,  
 $\text{not n}(\text{P2}, \text{K})$ .

$\text{n}(\text{P}, \text{K}) :-$

$\text{navstivil}(\_, \text{P}, \text{K})$ .

$\text{vypil\_nelubi}(\text{P}) :-$

$\text{navstivil}(\text{I}, \text{P}, \_)$ ,  
 $\text{vypil}(\text{I}, \text{A}, \_)$ ,  
 $\text{not lubi}(\text{P}, \text{A})$ .

Relačná algebra:

$\text{navstivil\_nenavstivil} := \pi_{\text{P1}, \text{P2}} ($

$((\rho_{\text{P1}, \text{Krcma}} \pi_{\text{Pijan}, \text{Krcma}} (\text{navstivil})) \times (\rho_{\text{P2}} \pi_{\text{Pijan}} (\text{navstivil}))) \triangleright (\rho_{\text{P2}, \text{Krcma}} \pi_{\text{Pijan}, \text{Krcma}} (\text{navstivil}))$

)

$\text{vypil\_nelubi} := \pi_{\text{Pijan}} ((\text{navstivil} \bowtie \text{vypil}) \triangleright \text{lubi})$

/\* answer \*/

$(\rho_{\text{P1}} \pi_{\text{Pijan}} (\text{navstivil}) \times \rho_{\text{P2}} \pi_{\text{Pijan}} (\text{navstivil})) \triangleright \rho_{\text{P1}, \text{P2}} (\text{navstivil\_nenavstivil}) \triangleright \rho_{\text{P2}, \text{P1}} (\text{navstivil\_nenavstivil}) \triangleright$   
 $\rho_{\text{P1}} (\text{vypil\_nelubi}) \triangleright \rho_{\text{P2}} (\text{vypil\_nelubi})$

b) Sformulujte bezpečný dotaz v Datalogu **(6)** a SQL **(6)** na dvojice [P, K] také, že pijan P navštívil krčmu K aspoň 100-krát a při každé své návštěvě K vypil nanajvýš 7 druhů alkoholu.

Datalog:

answer(P, K) :-

```
    subtotal(navstivil(I, P, K), [P, K], [C = count(I)]),  
    C >= 100,  
    \+ niekedy_vypil_viac(P, K).
```

niekedy\_vypil\_viac(P, K) :-

```
    navstivil(I, P, K),  
    subtotal(v(I, A), [I], [C = count(A)]),  
    C > 7.
```

v(I, A) :-

```
    vypil(I, A, _).
```

SQL:

with niekedy\_vypil\_viac as

```
(  
select n.Pijan, n.Krcma  
from navstivil n, vypil v  
where n.Idn = v.Idn  
group by n.Pijan, n.Krcma, v.Idn  
having count(distinct v.Alkohol) > 7  
)
```

select n.Pijan, n.Krcma

from navstivil n

where not exists (

```
    select *
```

```
    from niekedy_vypil_viac vv
```

```
    where n.Pijan = vv.Pijan and n.Krcma = vv.Krcma)
```

group by n.Pijan, n.Krcma

having count(n.Idn) >= 100

2. Uvažujte relačnú schému  $r(A, B, C, D, E, F, G, H)$  s funkčnými závislosťami  
 $AF \rightarrow G, AD \rightarrow B, ACH \rightarrow BE, BCD \rightarrow GH, BDG \rightarrow A, BG \rightarrow E, DG \rightarrow F, E \rightarrow BD$

a) Rozhodnite či  $r$  je v tretej normálnej forme. Odpoveď ÁNO resp. NIE zdôvodnite. (6)

Definícia: Relačná schéma  $[r, F]$  je v tretej normálnej forme, ak pre každú platnú kánonickú funkčnú závislosť (ktorá je dôsledkom závislostí  $F$ )  $X \rightarrow Y$  platí, že  $X$  je nadkľúč  $r$  alebo  $Y$  je časťou niektorého kľúča  $r$ .

Nájdime všetky kľúče  $r$ :

ABCDEF~~G~~H

-A: BCDEF~~G~~H

-B: CDEF~~G~~H

-D: CEF~~G~~H

-E: CF~~G~~H

+E: **CE**

+D: CDF~~G~~H

+B: BCDF~~G~~H

-D: BC~~F~~GH

-F: BC~~G~~H

-G: BC~~H~~

+G: **BCG**

+F: BC~~F~~H

+D: **BCD**

+A: ABC~~D~~F~~G~~H

-B: AC~~D~~F~~G~~H

-D: AC~~F~~GH

-F: AC~~G~~H

-G: **ACH**

+F: AC~~F~~G

+D: **ACD**

+B: **ABCF**

ABCF, ACD, ACH, BCD, BCG, CE sú kľúče v  $r$  (iné nie sú). Každý atribút  $r$  je časťou niektorého kľúča  $r$ . Z toho vyplýva, že **ÁNO,  $r$  je v tretej normálnej forme.**

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

V  $r(A, B, C, D, E, F, G, H)$  platí  $AF \rightarrow G$ , pričom  $AF$  nie je nadkľúč  $r$ , takže  $r$  nie je v BCNF.

Dekomponujme  $r$  na

$r_1(A, F, G)$ ,

$r_2(A, B, C, D, E, F, H)$ .

$r_1(A, F, G)$  je v BCNF.

V  $r_2$  platí  $AD \rightarrow B$ , pričom  $AD$  nie je nadkľúč  $r_2$ , takže  $r_2$  nie je v BCNF.

Dekomponujme  $r_2$  na

$r_3(A, B, D)$ ,

$r_4(ACDEFH)$ .

$r_3(A, B, D)$  je v BCNF.

V  $r_4$  platí  $E \rightarrow D$ , pričom  $E$  nie je nadkľúč  $r_4$ , takže  $r_4$  nie je v BCNF.

Dekomponujme  $r_4$  na

$r_5(D, E)$ ,

$r_6(ACEFH)$ .

$r_5(D, E)$  je v BCNF.

V  $r_6$  hľadáme platné kánonické funkčné závislosti s 3 atribútmi  $r_6$  na ľavej strane, kde ľavá strana nie je nadkľúč (iné funkčné závislosti s 3 atribútmi na ľavej strane nemôžu porušiť BCNF). Systematicky vypočítame uzávery všetkých trojíc atribútov, ktoré prichádzajú do úvahy:  $\{ACF\}^* = \{ACFG\}$ ,  $\{CFH\}^* = \{CFH\}$ ,  $\{EFH\}^* = \{BDEFH\}$ . Keďže nové atribúty z uzáverov nie sú v  $r_6$ , žiadna platná funkčná závislosť s 3 atribútmi na ľavej strane neporušuje BCNF.

V  $r_6$  hľadáme platné kánonické funkčné závislosti s 2 atribútmi  $r_6$  na ľavej strane, kde ľavá strana nie je nadkľúč. Nezaujímajú nás dvojice, ktoré sú podmnožinami trojíc, ktoré sme už overili vyššie. Žiadne také funkčné závislosti v  $r_6$  neplatia, takže  $r_6(A, C, E, F, H)$  je v BCNF.

Bezstratovosť spojenia výslednej dekompozície môžeme overiť algoritmom chase:

	A	B	C	D	E	F	G	H
<b>AFG</b>	a1					a6	a7	
<b>ABD</b>	a1	a2		a4				
<b>DE</b>				a4	a5			
<b>ACEFH</b>	a1	a2 <sub>3: AD→B</sub>	a3	a4 <sub>2: E→D</sub>	a5	a6	a7 <sub>1: AF→G</sub>	a8

**Bezstratová BCNF dekompozícia (niektoré funkčné závislosti sú zlomené, napr.  $E \rightarrow B$ ):**

**(A, F, G),  $AF \rightarrow G$**

**(A, B, D),  $AD \rightarrow B$**

**(D, E),  $E \rightarrow D$**

**(A, C, E, F, H),  $AF \rightarrow G$ ,  $ACH \rightarrow E$**

3. Uvažujte relácie  $r(A, B, C)$  a  $s(A, B, D)$  (bez duplikátov a NULL hodnôt) a výpočet relácie  $p(C)$  v relačnej algebre:

$$p := \pi_C(r \bowtie_{C=D} s) \cup (\pi_C(r) - \pi_C(\pi_{s.A, s.B, r.C}(r \times s) - r)).$$

a) Definujte v Datalogu charakteristický predikát  $p(\cdot)$  pre reláciu  $p$ . **(6)** (Relácie interpretujte ako predikáty.)

$p(C)$  :-

$r(\_, \_, C),$   
 $s(\_, \_, C).$

$p(C)$  :-

$r(\_, \_, C),$   
 not counterex(C).

counterex(C) :-

$r(\_, \_, C),$   
 $s(A, B, \_),$   
 not  $r(A, B, C).$

b) Vypočítajte reláciu  $p(C)$  pre

$r(A, B, C) = \{[0, 0, 1], [0, 0, 4], [0, 0, 2], [0, 1, 2], [0, 1, 4], [0, 2, 3], [1, 0, 2], [1, 1, 1], [1, 1, 4]\},$

$s(A, B, D) = \{[[0, 0, 0], [1, 1, 1]]\}$  **(6)**

Budeme počítat' podľa uvedeného výrazu v relačnej algebre:

$$\pi_C(r \bowtie_{C=D} s) = \{1\}$$

$$\pi_{s.A, s.B, r.C}(r \times s) = \{[0, 0, 1], [0, 0, 2], [0, 0, 3], [0, 0, 4], [1, 1, 1], [1, 1, 2], [1, 1, 3], [1, 1, 4]\}$$

$$\pi_{s.A, s.B, r.C}(r \times s) - r = \{[0, 0, 3], [1, 1, 2], [1, 1, 3]\}$$

$$\pi_C(\pi_{s.A, s.B, r.C}(r \times s) - r) = \{2, 3\}$$

$$\pi_C(r) - \pi_C(\pi_{s.A, s.B, r.C}(r \times s) - r) = \{1, 4\}$$

$$p(C) = \pi_C(r \bowtie_{C=D} s) \cup (\pi_C(r) - \pi_C(\pi_{s.A, s.B, r.C}(r \times s) - r)) = \{1, 4\}$$

4. Uvažujte veľkú reláciu (s veľa záznamami)  $r(A, B, C, D, E, F, G, H)$ , nad ktorou je vytvorených 8 indexov.

Indexy boli vytvorené SQL príkazmi

```
create index idx_a on r(A) using btree;
```

...

```
create index idx_h on r(H) using btree;
```

a) Predpokladajte, že relácia  $r$  sa používa v databázovej aplikácii, v ktorej žiaden dotaz neprofituje z indexov na atribútoch  $C, D, E, F, G, H$ . Popíšte situáciu, v ktorej prítomnosť indexov na týchto atribútoch môže výrazne znížiť efektivitu aplikácie. Vysvetlite dôvod zníženia efektivity. **(6)**

Efektivita sa môže výrazne znížiť pri intenzívnej aktualizácii databázy, lebo okrem dát v relácii  $r$  treba aktualizovať tiež všetky indexy pre  $r$ .

b) Uveďte príklad konkrétneho SQL dotazu nad reláciou  $r$ , ktorého vykonávanie sa môže urýchliť, keď sa na implementáciu indexu namiesto B-stromu na atribúte  $A$  použije hashovaný index. Vysvetlite. **(6)**

Príkladom je selekcia na konštantnú hodnotu atribútu  $A$ , v SQL

```
select A from r where A = <konštanta>
```

kde <konštanta> je konštantná hodnota príslušného dátového typu.

Pri B-tree indexe je zložitosť vyhľadávania prvého záznamu výsledku úmerná hĺbke B-stromu. Pri hashovanom indexe je táto zložitosť konštantná (za predpokladu dobrej hashovacej funkcie).