Ispit iz Baza podataka

(50 bodova) 8. rujna 2017.

Zadaci **1 - 6** odnose se na bazu podataka **restoran** koja je prikazana na **slici 1**. Na slici **nisu prikazane** sve n-torke sadržane u relacijama.

namirnica

sifNamirnica	*nazNam*
1	Rajčica
2	Češnjak
3	Škamp
4	Peršin
5	Crveni luk
6	Maslinovo ulje
7	T-Bone steak

jelo

sifJelo	*nazJelo*	*slozenost*	*trajanje*	*sifTip*	*vrZadNar*
1	Škampi na buzaru	3	20	1	2017-08-29 10:52:48.1542
2	Šalša od rajčica	3	35	1	2017-08-31 08:53:25.719
3	Škampi na žaru	4	10	2	2017-08-30 10:53:25.719
4	Salata od rajčice	2	5	3	<null>

sastav

sifJelo	***sifNamirnica***
1	1
1	2
1	3
1	4
1	6
4	1

tipJelo

sifTip	*nazTip*
1	Glavno jelo
2	Prilog
3	Salata
4	Predjelo

Slika 1.

Baza podataka **restoran** sadrži podatke o sastavima i tipovima jela u jednom restoranu. Relacija **jelo** sadrži podatke o jelima, gdje se uz naziv jela bilježi **složenost** pripreme jela (1-5), **trajanje** pripreme u minutama, tip jela, te vrijeme zadnje narudžbe (vrZadNar, može biti NULL). Sastav jela je evidentiran pomoću relacija **sastav** i **namirnica**.

U zadacima **1** - **3** napisati **po jednu SQL naredbu** kojom će se obaviti sljedeće:

1. **(5 bodova)** Jelima koja zadovoljavaju barem jedan od sljedeća dva uvjeta povećati složenost za jedan:
   - Priprema traje više od sat vremena
   - Priprema traje više od 35 minuta i sastoji se od 5 ili više namirnica

   Imajte na umu da je maksimalna složenost 5, tj. ne povećavati onim jelima koja su već maksimalne složenosti.

2. **(6 bodova)** Ispisati naziv namirnice i broj pojavljivanja za one namirnice koje se pojavljuju manje od tri puta u jelima koja su naručivana u proteklih godinu dana. U obzir uzeti samo glavna jela i priloge. Ispisati i namirnice koje se uopće ne pojavljuju u opisanim jelima.
   Zadatak riješiti bez upotrebe podupita.

3. **(6 bodova)** Ispisati (samo jednom) šifre jela u kojima se zajedno koristi:
   - bilo kakav luk (npr, 'Crveni **luk**', 'Ljubičasti **luk**', '**Luk** srebrenac', …)
   - češnjak.

   Zadatak riješiti bez upotrebe podupita!

4. **(7 bodova)** Napisati naredbe kojima će se u tablicu jelo dodati redundantni atribut **brNamirnica**, te vrijednost atributa postaviti na ispravnu vrijednost. Potom napisati naredbe kojima će se osigurati da taj podatak uvijek bude ažuran. Možete pretpostaviti da je nad tablicom **sastav** zabranjena operacija izmjene. Ne dopustiti da se **postojećim** jelima obrišu svi sastojci, tj. da imaju nula sastojaka (dojaviti iznimku s odgovarajućom porukom).
   Napomena: primjer korištenja naredbe alter table: `ALTER TABLE sastav ADD kolicina decimal(5,2);`

5. **(6 bodova)** Korisnik **postgres** kreirao je bazu podataka **restoran** i relacije u njoj sljedećim SQL naredbama:

```
CREATE DATABASE restoran;
CREATE TABLE jelo (...); CREATE TABLE namirnica (...);  ...
REVOKE CONNECT ON DATABASE restoran FROM PUBLIC;
REVOKE ALL     ON SCHEMA public FROM PUBLIC;
REVOKE ALL ON ALL TABLES IN SCHEMA public FROM PUBLIC;
```

   a) Kreirati ulogu *kuhar* s mogućnošću uspostavljanja SQL sjednice i pri tome koristiti autentifikaciju pomoću lozinke. Korisniku za inicijalnu lozinku postaviti "promijeni.me".
   b) Ulozi kuhar omogućiti pregled relacija jelo, tipJelo i namirnica, te izmjenu atributa jelo.vrZadNar.
   c) Kreirati ulogu *defSastav i* omogućiti joj unos i brisanje sastava jela, ali samo za jela koja nisu naručivana u tekućoj kalendarskoj godini. Putem uloge defSastav omogućiti ulozi kuhar definiranje sastava spomenutih jela.
   d) Ostvariti (ako već nije) da korisnik kuhar može dohvatiti reducirane podatke u relaciji sastav (iz c) dijela zadatka) tako da koristi upravo naziv „sastav", npr. `SELECT * FROM sastav` treba korisniku kuhar vratiti samo sastave jela koja „nisu naručivana u tekućoj kalendarskoj godini".

6. (4 boda) Za sve relacije u bazi restoran su definirani primarni ključevi (naznačeni na slici 1, ključ relacije sastav je kreiran navedenim redom: PRIMARY KEY(sifJelo, sifNamirnica)).
Obavlja se sljedeći upit, pri čemu optimizator upita raspolaže navedenim statističkim informacijama:

`SELECT *`
`  FROM jelo`
`  NATURAL JOIN sastav`
`  NATURAL JOIN namirnica`
`WHERE slozenost < 4`
`  AND sifNamirnica = 11`
	`N(jelo) = 9.000`
	`V(slozenost, jelo) = 5`
	
	`N(sastav)= 5.000`
	`V(sifNamirnica, sastav) = 500`
	
	`N(namirnica) = 500`

Nacrtati moguća stabla izvođenja upita (za različite moguće redoslijede spajanja relacija) s navedenim procjenama broja n-torki u međurezultatima i načinima pristupa podatcima, te obrazložiti koje stablo (redoslijed spajanja) će odabrati optimizator.

7. (4 boda) Prikazan je odsječak sadržaja relacije ***tipJelo*** u trenutku započinjanja naredbe {1}. Odredite rezultate naredbi {5}, {7}, i {8} ako su u trenutku započinjanja transakcija A, B i C aktivne transakcije s identifikatorima 10 i 12, a zadnja potvrđena transakcija ona s identifikatorom 11. Pretpostavite da do kraja naredbe {10}

- osim transakcije A niti jedna druga transakcija ne mijenja sadržaj relacije ***tipJelo***
- su transakcije 10 i 12 ostale aktivne
- nije aktiviran proces koji "čisti" stare n-torke (*garbage collector*)

tipJelo

xmin	***xmax***	***sifTip***	***nazTip***
1	0	4	Predjelo
2	10	5	Dessert
10	0	5	Desert
12	0	6	Umak

	Transakcija A T_{id}=13		**Transakcija B T_{id}=14**		**Transakcija C T_{id}=15**
{1}	`BEGIN TRANSACTION;`	{2}	`BEGIN TRANSACTION;`	{3}	`BEGIN TRANSACTION;`
{4}	`INSERT INTO tipJelo` `  VALUES (7, 'Koktel');`		`SET TRANSACTION ISOLATION` `LEVEL SERIALIZABLE;`		

{5}	`SELECT xmin, xmax, *`
	`  FROM tipJelo`
	`WHERE sifTip BETWEEN 4 AND 7;`				

{6}	`COMMIT TRANSACTION;`	{7}	`SELECT xmin, xmax, *`
			`  FROM tipJelo`
			`WHERE sifTip BETWEEN 4 AND 7;`
				{8}	`SELECT xmin, xmax, *`
					`  FROM tipJelo`
					`WHERE sifTip BETWEEN 4 AND 7;`

		{9}	`COMMIT TRANSACTION;`	{10}	`COMMIT TRANSACTION;`

8. (8 bodova)
Potrebno je izraditi model podataka tvrtke za popravak i prodaju rabljenih automobila. Kupci rabljenih automobila mogu biti pravne ili fizičke osobe. Obje vrste osoba identificiraju se OIB-om. Pored toga, za pravnu osobu bilježi se naziv i matični broj, a za fizičku osobu ime i prezime osobe. Automobil je identificiran tvorničkim brojem (ne postoje dva automobila s istim tvorničkim brojem). Za automobil treba evidentirati godinu proizvodnje i proizvođača, a za proizvođača šifru i naziv.
Tvrtka evidentira podatke o svojim zaposlenicima (fizičke osobe) koji mogu biti mehaničari ili prodavači, ili oboje. Mehaničarima se dodatno evidentira razina osposobljenosti (1-5), a prodavačima broj licence. Mehaničari na automobilima obavljaju niz tipskih popravaka. Za tipske popravke se evidentira šifra, naziv i složenost (broj 1-5). Prilikom rada na automobilu stvara se radni nalog kojim se, pored datuma i jedinstvenog broja radnog naloga, evidentira koje tipske popravke će koji mehaničar obavljati na nekom automobilu. Na nekom radnom nalogu, jedan tip popravka obavlja samo jedan mehaničar. Kupci mogu kupovati iste automobile više puta (ali ne na isti dan). Za prodani automobil pored iznosa i datum prodaje, evidentira se zaposlenik (samo jedan) koji je prodaju realizirao. Nacrtati ER model baze podataka, navesti sheme entiteta i sheme veza (označiti ključeve). Svaki entitet (osim slabih entiteta) opisuje se isključivo vlastitim atributima. Nužno je da sve sheme zadovoljavaju 3NF.

9. (4 boda) Relacija **NAMIRNICA (sifNamirnica, nazNam)** sadrži 7 n-torki prikazanih na slici 1. Nacrtati dva B+ stabla reda 5 za atribut **nazNam** tako da popunjenost stabla bude **u jednom slučaju minimalna**, a u **drugom slučaju maksimalna**. Nacrtati sve relevantne kazaljke. Odrediti koliko je u kojem slučaju potrebno obaviti UI operacija za dohvaćanje jednog zapisa traženog prema vrijednosti atributa **nazNam**.

RJEŠENJA

1.

```
UPDATE jelo
  SET slozenost = slozenost + 1
  WHERE slozenost < 5
    AND ( trajanje > 60
         OR (      trajanje > 35
              AND (SELECT COUNT(*) FROM sastav WHERE sifJelo = jelo.sifJelo) >= 5
            )
         )
```

2.

```
   SELECT nazNam, count(tipJelo.sifTip) as br
     FROM jelo join tipJelo
                 ON jelo.sifTip = tipJelo.sifTip
                AND nazTip IN ('Glavno jelo', 'Prilog')
                and vrZadNar > current_timestamp - '1 YEAR'::interval
               natural join sastav
               NATURAL RIGHT JOIN namirnica

   group by nazNam
   having count(*) < 3
```

3.

```
SELECT distinct sastavLuk.sifJelo
FROM  sastav sastavLuk
       JOIN namirnica n1
         ON sastavLuk.sifNamirnica = n1.sifNamirnica
        AND lower(n1.nazNam) LIKE '%luk%'
     JOIN sastav sastavCesnjak
       ON sastavLuk.sifJelo = sastavCesnjak.sifJelo
     JOIN namirnica n2
         ON sastavCesnjak.sifNamirnica = n2.sifNamirnica
        AND lower(n2.nazNam) LIKE '%češnjak%'
```

4.

```
alter table jelo add brSastojaka smallint;

update jelo set brSastojaka = (select count(*) from sastav where sifJelo =
jelo.sifJelo);

CREATE FUNCTION updJeloBrSast()
  RETURNS TRIGGER AS $$
DECLARE
  p_br SMALLINT;
  p_sifJelo DECIMAL(8,2);
BEGIN
 SELECT CASE WHEN (TG_OP = 'DELETE') THEN OLD.sifJelo
             ELSE NEW.sifJelo
        END
   INTO p_sifJelo;

  SELECT COUNT(*) INTO p_br
    FROM sastav
   WHERE sifJelo = p_sifJelo;

  IF (p_br = 0) THEN
    RAISE EXCEPTION 'Ne možete obrisati posljedni sastojak!';
  ELSE
    UPDATE jelo
     SET brSastojaka = p_br
     WHERE sifJelo = p_sifJelo;
  END IF;

  RETURN NEW;  -- OK, delete može vratiti NULL u row level after triggeru.
END;
$$ LANGUAGE plpgsql;
```

```
CREATE TRIGGER sastavInsDel
  AFTER INSERT OR DELETE ON sastav
  FOR EACH ROW
    EXECUTE PROCEDURE updJeloBrSast();
```

5.

a)

```
CREATE ROLE kuhar  LOGIN PASSWORD 'promijeni.me';
--ove dvije naredbe je potrebno obaviti, ali mogu biti i dio rješenja zadatka pod b)
GRANT CONNECT ON DATABASE restoran TO kuhar;
GRANT USAGE ON SCHEMA PUBLIC TO kuhar;
```

b)

```
grant select on jelo, tipJelo, namirnica to kuhar ;
grant update (vrZadNar) on jelo to kuhar;
```

c)

```
create role defSastav;
create view sastavOvaGodina as
SELECT *
 FROM sastav
 WHERE sifJelo NOT IN (SELECT sifJelo FROM jelo WHERE extract(YEAR FROM vrZadNar) <
extract(YEAR FROM current_timestamp))
with check option;

grant select, insert, delete on sastavOvaGodina to defSastav;

grant defSastav to kuhar;
insert into sastavOvaGodina values (...)
```

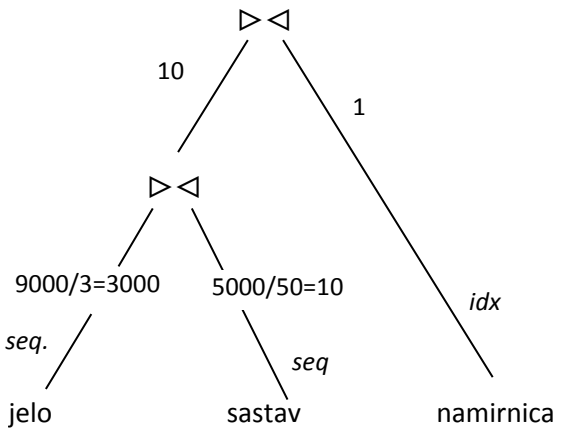
d)

```
create schema kuhar
create view kuhar.sastav as
    select * from sastavOvaGodina;
grant usage on schema kuhar to kuhar;
grant select, insert, delete on kuhar.sastav to defSastav;
```

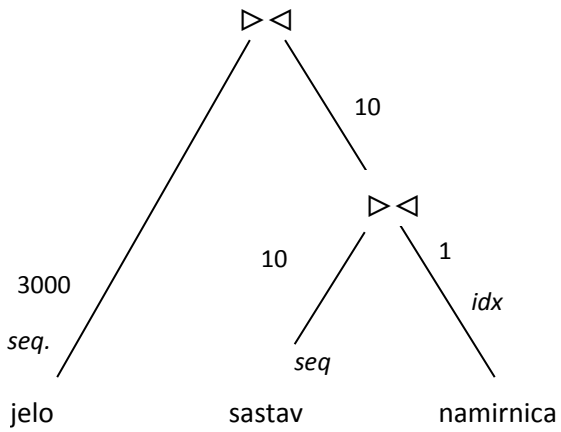
6.

Najbolje (b), ali u redu je i (a).

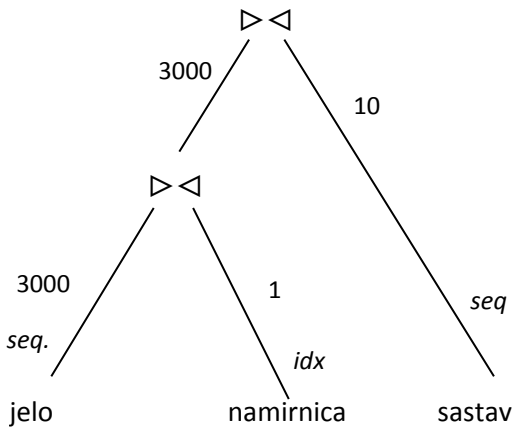
a)



b)



c)



7.

{5}

xmin	xmax	sifJelo	nazJelo
1	0	4	Predjelo
2	10	5	Dessert
13	0	7	Koktel

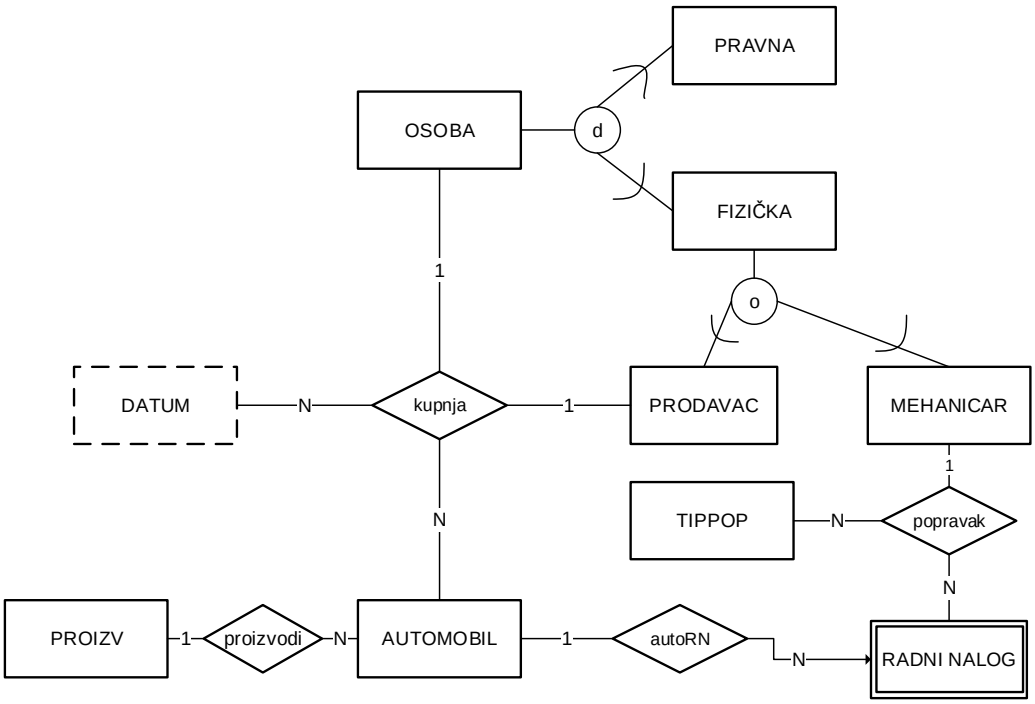
{7}

xmin	xmax	sifJelo	nazJelo
1	0	4	Predjelo
2	10	5	Dessert

{8} (isto što i 5) :

xmin	xmax	sifJelo	nazJelo
1	0	4	Predjelo
2	10	5	Dessert
13	0	7	Koktel

8.



OSOBA
OIB

PRAVNA
OIB
MBR
naziv
K2 = {MBR}

FIZIČKA
OIB
ime
prezime

PRODAVAC
OIB
brLicence

MEHANICAR
OIB
razOsposob

TIPPOP
sifTipPop
naziv
slozenost

kupnja
tvBroj
datKup
OIBKupac
OIBProdavac
iznos

proizvodi
tvBroj
sifProizv

popravak
brRNalog
sifTipPop
OIB

autoRN
brRNalog
tvBroj

AUTOMOBIL
tvBroj
godProizv

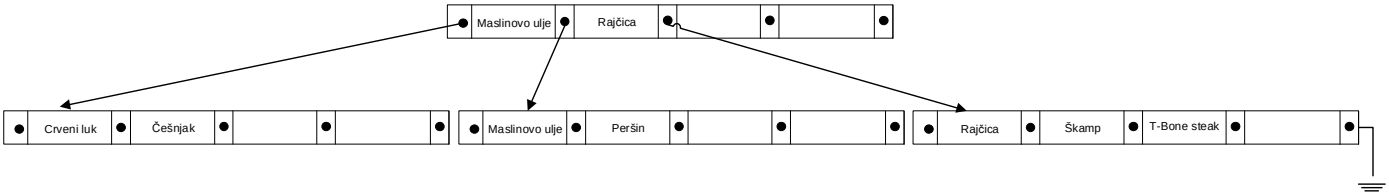
RADNINALOG
brRNalog
datRN

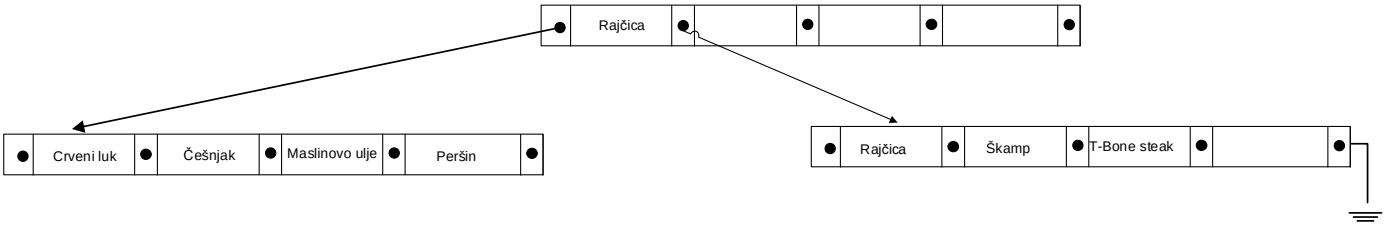
PROIZV
sifProizv
nazProizv

9.

- **Minimalno popunjeno stablo:**
  - list: minimalno 2
  - interni čvor minimalno: 3, korijen 2

UI = 2 + 1



- **Maksimalno popunjeno stablo:**
  - list: maksimalno 4
  - interni čvor: maksimalno 5

UI = 2 + 1