

Završni ispit iz Baza podataka (40 bodova)

20. lipnja 2017.

Zadaci 1 - 6 odnose se na bazu podataka prikazanu na **slici 1**. U zadacima se podrazumijeva korištenje SUBP-a PostgreSQL. Baza podataka služi za evidenciju podataka o lokalnim crossfit natjecanjima koja uključuju vježbe sa zadanim brojem ponavljanja. Za natjecanje se unaprijed propisuje broj krugova u kojima je potrebno odraditi zadane vježbe. Za svaku vježbu zadan je broj ponavljanja i ukupna težina tereta (tablica **natVjezba**). Mjeri se i bilježi vrijeme (**natOsobaRez.vrijeme**) potrebno za obavljanje svake zadane vježbe u svakom krugu. Natjecatelju koji ne uspije odraditi vježbu u cijelosti, kao vrijeme se bilježi vrijednost NULL i biva diskvalificiran iz ostatka natjecanja. Atribut **natOsoba.brOdrVj** redundantno čuva broj odradenih vježbi natjecatelja na nekom natjecanju. Ključevi relacija su podcrtani. Na slici **nisu** prikazane sve n-torke sadržane u relacijama.

osoba					nat		natOsoba		
<u>sifQ</u>	ime	prezime	<u>datRod</u>	login	<u>datNat</u>	nazNat	<u>datNat</u>	<u>sifQ</u>	<u>brOdrVj</u>
100	Mia	Toš	15.04.1974	mtos	1.2.2016	Reebok Fem	01.05.2017	100	10
110	Ana	Pecci	10.02.1985	apecci	2.3.2017	Crossfit ZG	01.05.2017	110	15
130	Iva	Tolj	17.04.1990	atolj	1.5.2017	Reebok Challenge	01.05.2017	120	25
...

vjezba		natVjezba						natOsobaRez				
<u>sifVj</u>	opisVj	<u>datNat</u>	<u>rbr</u> <u>Krug</u>	<u>rbrVj</u>	<u>sifVj</u>	<u>brPon</u>	<u>tezina</u>	<u>sifQ</u>	<u>datNat</u>	<u>rbr</u> <u>Krug</u>	<u>rbrVj</u>	<u>vrijeme</u>
1	Trzaj	01.05.2017	1	1	10	20	40	100	01.05.2017	1	1	00:01:10
2	Nabačaj-izbačaj	01.05.2017	2	1	20	12	50	110	01.05.2017	1	1	00:01:50
3	Mrtvo dizanje	12.06.2017	1	2	10	25	80	110	01.05.2017	1	2	00:02:20
...	130	01.05.2017	2	3	NULL

Slika 1.

U zadacima 1 i 2 napisati **po jednu SQL naredbu** kojom će se obaviti sljedeće:

- (4 boda)** Ispisati naziv natjecanja, te za svaki krug natjecanja redni broj kruga i broj natjecatelja koji su diskvalificirani u tom krugu (nula - 0 ako u tom krugu nije diskvalificiran nijedan natjecatelj). Zapisе poredati po nazivu natjecanja abecedno, a zatim po broju odustajanja silazno.

- (5 bodova)** Za natjecanja održana u svibnju tekuće godine ispisati trend (pada/jednak/raste/nema) pojedine vježbe u pogledu broja ponavljanja u uzastopnim krugovima. Npr. ako vježba opisa „Trzaj“ u 2. krugu zahtijeva 10 ponavljanja, a u 3. krugu 15 ponavljanja tada je trend „raste“ itd. Ako se određena vježba ne pojavljuje u sljedećem krugu za trend ispisati „nema“.

Izgled ispisa:

<u>datNat</u>	<u>sifVj</u>	<u>opisVj</u>	<u>krug1</u>	<u>br1</u>	<u>krug2</u>	<u>br2</u>	<u>trend</u>
01.05.2017	2	Nabačaj-izbačaj	1	20	2	10	pada
01.05.2017	1	Trzaj	2	10	3	15	raste
15.05.2017	3	Mrtvo dizanje	2	10	3	10	jednak
15.05.2017	1	Trzaj	1	20	2		nema

- (6 bodova)** Administrator baze podataka je kreirao bazu podataka i relacije u njoj sljedećim SQL naredbama:

```
CREATE DATABASE crossfit;
CREATE TABLE osoba (...); CREATE TABLE nat (...); ...
REVOKE CONNECT ON DATABASE crossfit FROM PUBLIC;
REVOKE ALL ON SCHEMA public FROM PUBLIC;
REVOKE ALL ON ALL TABLES IN SCHEMA public FROM PUBLIC;
```

ovlasti	
<u>login</u>	<u>datNat</u>
apecci	01.05.2017
apecci	15.05.2017
itos	01.06.2017
...	...

Vrijednost atributa **ovlasti.login** jednaka je korisničkom imenu korisnika koji je uspostavio sjednicu (SESSION_USER). Relacijom ovlasti su definirana natjecanja (**datNat**) s čijim podacima pojedini korisnik smije raditi. Potrebno je:

- stvoriti ulogu *adminNat* i shemu *adminNat* čiji je vlasnik uloga *adminNat*
- koristeći shemu *adminNat* omogućiti ulozi *adminNat* pregledavanje, izmjenu, brisanje i unos podatka u relaciju **natOsoba** u skladu s ovlastima u tablici **ovlasti**, a pri tom se odnose na podatke s natjecanja održanih u zadnjih šest mjeseci. Nije dozvoljeno stvarati objekte u shemi *public*.
Dodatno, ulozi *adminNat* treba dodijeliti ovlast pregledavanja svih podataka o osobama (relacija **osoba**).
- korisniku *apecci* dodijeliti ovlasti za uspostavu korisničke sjednice te ovlasti za rad u ulozi *adminNat*
- napisati SQL naredbe koje korisnik *apecci*, nakon što uspostavi korisničku sjednicu, treba obaviti da bi pregledao podatke iz relacije **osoba** i podatke, za koje ima ovlasti, iz relacije **natOsoba**.

- (6 bodova)** Napisati niz SQL naredbi za kreiranje svih potrebnih objekata kojima će se pri unosu n-torke u relaciju **natOsobaRez** održavati ispravnom vrijednost atributa **natOsoba.brOdrVj** (za svaku odradenu vježbu povećati broj odradenih vježbi za 1). Dodatno, unos zapisa u kojem je vrijeme po jednom ponavljanju (ukupno vrijeme izvođenja vježbe podijeljeno s brojem ponavljanja) dulje od jedne minute treba spriječiti, a korisniku dojaviti sljedeću pogrešku: **Pogreška: trajanje jednog ponavljanja iznosi: 00:03:44. Rezultat treba diskvalificirati!**

Održavanje konzistentnosti navedenog atributa pri obavljanju ostalih akcija u relaciji **natOsobaRez** nije potrebno implementirati.

5. (5 bodova) Sve relacije sa slike 1 su kreirane u bazi podataka. Pri tom su definirani integriteti ključa i referencijski integriteti.

Izvodi se upit:	Optimizator raspolaže sljedećim podacima	
<pre>SELECT * FROM natOsobaRez NATURAL JOIN natOsoba NATURAL JOIN osoba WHERE datNat = '01.05.2017' AND brOdrVj = 20</pre>	<p>N(osoba) = 500 N(natOsoba) = 15000 N(natOsobaRez) = 150 000</p>	<p>V(rbrKrug, natVjezba) = 10 V(rbrVj, natVjezba) = 5 V(brOdrVj, natOsoba) = 50 V(datNat, natOsoba) = 150 V(datNat, natOsobaRez) = 100</p>

Nacrtao stablo upita nakon provedene heurističke optimizacije. Dovoljno je nacrtati samo konačno stablo upita. Redoslijed spajanja relacija odrediti temeljem procjene broja n-torki u rezultatima spajanja. Navesti sve izraze prema kojima je obavljena procjena broja n-torki u međurezultatima i u konačnom rezultatu. U stablu upita naznačiti očekivani broj n-torki i očekivane metode pristupa podacima.

6. (4 boda) Prikazan je odsječak sadržaja relacije **vjezba** u trenutku započinjanja naredbe {1}. Odredite rezultate naredbi {4}, {5}, i {7} ako je u trenutku započinjanja transakcija A i B aktivna transakcija s identifikatorom 50, a zadnja potvrđena transakcija ona s identifikatorom 51. Pretpostavite da do kraja naredbi {6} i {8}

- izuzev transakcije A niti jedna druga transakcija ne mijenja sadržaj relacije **vjezba**
- je transakcija 50 ostala aktivna
- nije aktiviran proces koji "čisti" stare n-torke (garbage collector)

vjezba

xmin	xmax	sifVj	opisVj
10	0	1	Trzaj
15	35	2	Nabačaj-izbačaj
35	0	2	Nabačaj/izbačaj
20	50	3	Mrtvo dizanje
50	0	3	Dead lift
...			...

	Transakcija A T _{id} =52		Transakcija B T _{id} =53
{1}	BEGIN TRANSACTION;	{2}	BEGIN TRANSACTION;
{3}	DELETE FROM vjezba WHERE sifVj = 1;	{4}	SET TRANSACTION ISOLATION LEVEL SERIALIZABLE; SELECT xmin, xmax, * FROM vjezba WHERE sifVj BETWEEN 1 AND 3;
{5}	SELECT xmin, xmax, * FROM vjezba WHERE sifVj BETWEEN 1 AND 3;	{7}	SELECT xmin, xmax, * FROM vjezba WHERE sifVj BETWEEN 1 AND 3;
{6}	COMMIT TRANSACTION;	{8}	COMMIT TRANSACTION;

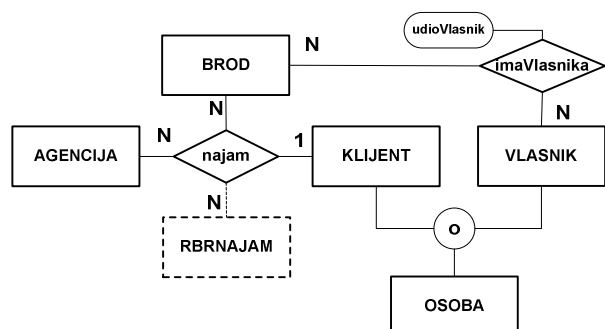
7. (5 bodova) Oblikovati ER model segmenta baze podataka za praćenje podataka o aktivnostima planinara i planinarskih društava. Evidentiraju se:

- Planinarska društva (šifra, naziv društva i godina osnutka).
- Planinarski domovi (šifra, naziv doma)
Planinarski dom pripada određenom planinarskom društvu koje se brine za njegovo održavanje. Neka planinarska društva održavaju više planinarskih domova.
- Planinari (šifra, ime i prezime te OIB).
Planinari su članovi planinarskih društava (jednog ili više njih) i svake godine plaćaju društvu, čiji su članovi, članarinu. Potrebno je evidentirati datum uplate za svaku kalendarsku godinu za koju je član platio članarinu društvu/društvima.
- Izleti planinarskih društava (datum polaska, trajanje u danima, opis, cijena, planinarsko društvo koje ga organizira, planinar-vodič izleta i planinari koji u izletu sudjeluju)
Izlet organizira jedno planinarsko društvo i ima jednog vodiča koji je i sam planinar. Planinarsko društvo može s istim datumom polaska organizirati više izleta. Za izlet se dodatno evidentira redni broj izleta, pri čemu određenog datuma polaska za svako planinarsko društvo redni brojevi izleta započinju ponovo s brojem jedan. Za isti datum i isto planinarsko društvo ne postoje dva izleta s istim rednim brojem.

Nacrtao 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**.

8. (5 bodova) Za zadani E-R model napisati naredbe za kreiranje ekvivalentnog relacijskog modela u obliku SQL naredbi za kreiranje relacija s ugrađenim opisima primarnih, alternativnih ključeva i općih pravila integriteta. Tipove podataka u naredbama nije potrebno navoditi.

AGENCIJA = sifAgencija, naziv
BROD = regOznaka, imeBrod
OSOBA = OIB, ime, prezime
KLIJENT = eMail, datumOd
VLASNIK = OIB



Rješenja:

1. 4 boda

```
SELECT nazNat, natVjezba.rbrKrug,
       COUNT(natOsobaRez.rbrVj) odustalo
FROM   natVjezba
NATURAL JOIN nat
LEFT OUTER JOIN natOsobaRez
  ON natOsobaRez.datNat = natVjezba.datNat
  AND natOsobaRez.rbrKrug = natVjezba.rbrKrug
  AND natOsobaRez.rbrVj = natVjezba.rbrVj
  AND vrijeme IS NULL
GROUP BY nazNat, natVjezba.rbrKrug
ORDER BY nazNat, odustalo DESC
```

2. 5 bodova

```
SELECT nv1.datNat, nv1.sifVj, vjezba.opisVj,
       nv1.rbrKrug Krug1, nv1.brPon br1,
       nv1.rbrKrug+1 krug2, nv2.brPon br2,
       CASE WHEN nv2.brPon IS NULL THEN 'nema'
            WHEN nv1.brPon < nv2.brPon THEN 'raste'
            WHEN nv1.brPon > nv2.brPon THEN 'pada'
            ELSE 'jednak'
       END
FROM   natVjezba nv1
NATURAL JOIN vjezba
LEFT OUTER JOIN natVjezba nv2
  ON nv1.datNat = nv2.datNat
  AND nv1.rbrKrug = nv2.rbrKrug-1
  AND nv1.sifVj = nv2.sifVj
WHERE  EXTRACT(MONTH FROM nv1.datNat) = 5
      AND EXTRACT(YEAR FROM nv1.datNat) = EXTRACT(YEAR FROM CURRENT_DATE)
```

3. 6 bodova

```
a)
CREATE ROLE adminNat;
CREATE SCHEMA adminNat AUTHORIZATION adminNat;

b)
CREATE VIEW adminNat.vNatOsoba AS
  SELECT *
    FROM natOsoba
   WHERE datNat IN (SELECT datNat FROM ovlasti WHERE login = SESSION_USER
                  AND datNat BETWEEN (CURRENT_DATE- '6 MONTH'::INTERVAL)
                  AND CURRENT_DATE)
  WITH CHECK OPTION;
GRANT SELECT, INSERT, UPDATE, DELETE ON adminNat.vNatOsoba TO adminNat;
GRANT USAGE ON SCHEMA public TO adminNat;
GRANT SELECT ON osoba TO adminNat;

b)
GRANT CONNECT ON DATABASE crossfit TO apecci;
GRANT adminNat TO apecci;

c)
SELECT * FROM osoba;
SELECT * FROM adminNat.vnatosoba;
ili
SET role adminNat;
SELECT * FROM vnatosoba;
```

4. (6 bodova)

```
--DROP TRIGGER insNatOsobaRez ON NatOsobaRez;
--DROP FUNCTION azurBrOdrVj();
CREATE FUNCTION azurBrOdrVj() RETURNS trigger AS $$
DECLARE
  p_trajelPonavljanje INTERVAL;
BEGIN
  IF NEW.vrijeme IS NULL THEN
```

```

RETURN NEW;
END IF;
SELECT NEW.vrijeme/natVjezba.brPon
INTO p_trajelPonavljanje
FROM natVjezba
WHERE datNat = NEW.datNat
AND rbrKrug = NEW.rbrKrug
AND rbrVj = NEW.rbrVj;
IF p_trajelPonavljanje > '00:01:00'::INTERVAL THEN
RAISE EXCEPTION 'Pogreška: trajanje jednog ponavljanja: %. Rezultat treba
diskvalificirati!', p_trajelPonavljanje;

ELSE
UPDATE natOsoba
SET brOdrVj = brOdrVj +1
WHERE datNat = NEW.datNat
AND sifo = NEW.sifo;
RETURN NEW;
END IF;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER insNatOsobaRez
AFTER INSERT ON natOsobaRez
FOR EACH ROW
EXECUTE PROCEDURE azurBrOdrVj();

```

5. (5 boda)

$N(\text{natOsobaRez1}) = N(\text{natOsobaRez}) / (V(\text{datNat}, \text{natOsobaRez}) = 150000/100 = 1500$
 $N(\text{natOsoba1}) = N(\text{natOsoba}) / (V(\text{datNat}, \text{natOsoba}) * V(\text{brOdrVj}, \text{natOsoba})) = (15000/150*50)=2$

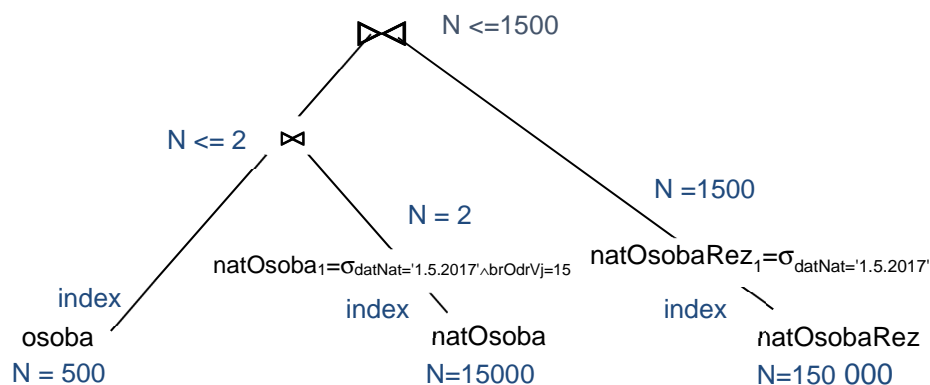
Procjena broja n-torki u međurezultatu za različite redoslijede spajanja:

$N(\text{natOsobaRez1} \bowtie \text{natOsoba1}) \leq N(\text{natOsobaRez1}) \leq 1500$

$N(\text{natOsobaRez1} \bowtie \text{osoba}) \leq N(\text{natOsobaRez1}) \leq 1500$

$N(\text{natOsoba1} \bowtie \text{osoba}) \leq N(\text{natOsoba1}) \leq 2$

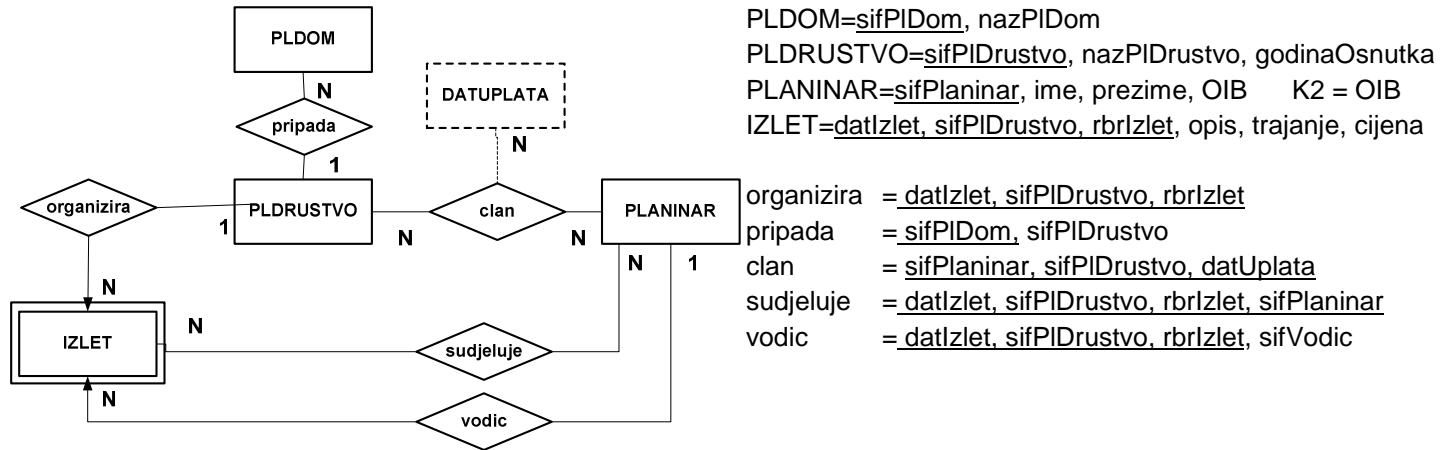
Redoslijed: 1. $N(\text{natOsoba1} \bowtie \text{osoba})$



6. (4 boda)

{ 4 }				{ 5 }				{ 7 }			
xmin	xmax	sifVj	opisVj	xmin	xmax	sifVj	opisVj	xmin	xmax	sifVj	opisVj
10	52	1	Trzaj	35	0	2	Nabačaj/izbačaj	10	52	1	Trzaj
35	0	2	Nabačaj/izbačaj	20	50	3	Mrtvo dizanje	35	0	2	Nabačaj/izbačaj
20	50	3	Mrtvo dizanje					20	50	3	Mrtvo dizanje

7. (5 bodova)



8. (5 bodova)

```
CREATE TABLE agencija (sifAgencija ... PRIMARY KEY, naziv ...);
CREATE TABLE brod (regOznaka ... PRIMARY KEY, imeBrod ...);
CREATE TABLE osoba (
    OIB ... PRIMARY KEY,
    ime ...,
    prezime ... );
CREATE TABLE klijent (
    eMail ... PRIMARY KEY,
    datumOd ...,
    OIB ... REFERENCES osoba(OIB));
CREATE TABLE vlasnik (
    OIB ... PRIMARY KEY REFERENCES osoba(OIB));
CREATE TABLE najam (
    eMailklijent ... REFERENCES klijent(eMail),
    regOznaka ... REFERENCES brod(regOznaka),
    sifAgencija ... REFERENCES agencija(sifAgencija),
    rbrNajam ...
    PRIMARY KEY (regOznaka, sifAgencija, rbrNajam)
);
CREATE TABLE imaVlasnika (
    OIBVlasnik ... REFERENCES vlasnik(OIB),
    regOznaka ... REFERENCES brod(regOznaka),
    udioVlasnik ...
    PRIMARY KEY (OIBVlasnik, regOznaka)
);
```