

Međuispit iz Baza podataka

24. travnja 2017.

Zadaci 1 - 5 i 9. odnose se na bazu podataka prikazanu na **slici 1**. Na slici **nisu** prikazane sve n-torke sadržane u relacijama.

Staza						PIDrustvo	
sifStaza	nazivStaza	tezina	duljina	sifPIDrustvo	datOdrzavanje	sifPIDrustvo	nazPIDrustvo
20	Miliči-Sv. Jure	5	5.0	1	01.03.2016	1	HPD "Sveti Jure Zagvoz
21	Prišivec-Oštrc	2	4.8	2	17.06.2015	2	HPD "Japetić"
22	Bliznec-Sljeme	3	7.6	1	12.04.2017	3	Planinarsko društvo "Kalnik"
24	Poučna Kalnik staza	2	5.0	1	01.07.2002
25	Škrbina draga – Rossijeva koliba	3	2.5	4	15.11.2006		
...		

Objekt			TipObjekt		StazaObjekt	
sifObjekt	nazivObj	sifTipObj	sifTipObj	opisTipObj	sifStaza	sifObjekt
11	Stari bunar	560	555	planinarski dom	20	11
12	Ruševna staja	575	560	bunar	20	12
13	PD Kozorog	555	575	ruševina	21	14
14	PD Runolist	555	580	spomenik
...		

Slika 1.

U bazi podataka se pohranjuju podaci o planinarskim stazama, planinarskim društvima koja ih održavaju te objektima koji se nalaze uz planinarske staze. Za staze se, pored šifre i naziva, evidentira procijenjena težina $\in \{1, 2, 3, 4, 5\}$, duljina u kilometrima, planinarsko društvo zaduženo za održavanje staze i zadnji datum održavanja. Podcrtani atributi predstavljaju ključ relacijske sheme.

1. (3 boda) Preinačite sljedeći SQL upit tako da ne koristi niti jedan podupit:

```
SELECT sifStaza, nazivStaza, tezina,
       (SELECT COUNT(sifObjekt)
        FROM stazaobjekt
        WHERE stazaobjekt.sifStaza = staza.sifStaza) AS brojObjekata
FROM staza
WHERE sifPIDrustvo IN (SELECT sifPIDrustvo
                       FROM PIDrustvo
                       WHERE nazivPIDrustvo LIKE '%HPD%')
AND (SELECT COUNT(sifObjekt)
     FROM stazaobjekt
     WHERE stazaobjekt.sifStaza = staza.sifStaza) > 2;
```

U zadacima 2 i 3 napisati **po jednu SQL naredbu** kojom će se obaviti sljedeće:

- (3 boda) Obrisati objekte koji se ne nalaze niti uz jednu stazu (StazaObjekt), a čiji tip sadrži niz slova "utvrda" (neovisno o malim i velikim slovima).
- (5 bodova) Za sva planinarska društva s ispravnim nazivom ispisati naziv, godinu te koliko su **različitih** staza, po godinama, održavali u zadnjih 5 godina (tekuća i prethodne četiri cijele godine). Ispravni su nazivi planinarskih društava koji ne sadrže (dvostruki) navodnik ili sadrže točno dva (dvostruka) navodnika. Obratno, neispravni su nazivi koji sadrže točno jedan (dvostruki) navodnik. U obzir uzeti samo staze uz koje se nalaze objekti (StazaObjekt). Podatke ispisati i za planinarska društva (ispravnog naziva) koja u zadnjih 5 godina nisu održavala niti jednu stazu. Rezultate poredati po nazivu društva uzlazno, te po godinama silazno. Riješiti bez upotrebe podupita.
- (3 boda) Napisati izraz relacijske algebre (**ne SQL upit!**) koji za svako planinarsko društvo i težine staza koje održava, ispisuje naziv društva, težinu staze, broj staza i ukupnu duljinu staza te težine. U obzir uzeti samo staze težine 4 ili 5. Za planinarska društva koja ne održavaju niti jednu stazu navedenih težina izraz treba ispisati vrijednost 0 za broj i NULL za ukupnu duljinu staza.
- (2 boda) Uz pretpostavku da su na **Slici 1**. prikazane sve n-torke u relacijama skicirati tablicu koja reprezentira rezultat sljedećeg izraza:

$\rho_{A(\text{opis, brojObjekata})} (\text{opisTipObj} \bowtie G \text{ COUNT (sifObjekt) (Objekt} \bowtie * \text{ tipObjekt) })$

6. (5 bodova) Planinarsko društvo ima bazu u kojoj čuva informacije o izletima svojih članova. Relacijska shema IZLETI ima sljedeće atribute:

OIB	<i>osobni identifikacijski broj planinara</i>	trajanjelzlet	<i>broj dana trajanja izleta</i>
imeP	<i>ime planinara</i>	sifPlanina	<i>šifra planine s vrhovima</i>
prezP	<i>prezime planinara</i>	nazivPlanina	<i>naziv planine</i>
pbrStan	<i>poštanski broj mjesta stanovanja planinara</i>	sifVrh	<i>šifra vrha</i>
nazivMjStan	<i>naziv mjesta stanovanja</i>	nazivVrh	<i>naziv vrha</i>
sifIzlet	<i>šifra izleta</i>	visinaVrh	<i>visina vrha</i>
datIzlet	<i>datum početka izleta</i>		

Izlet se organizira određenog datuma na određeni vrh. Izlet ima svoju šifru, datum i vrh koji je odredište izleta. Planinar može ići na više izleta. Planina može imati više vrhova.

Odrediti ključ relacijske sheme IZLETI tako da ona bude u 1NF, a zatim postupno normalizirati relacijsku shemu na 2NF i 3NF.

7. (2 boda) Zadana je relacijska shema $R = ABCDEFGH$ i skup funkcijskih zavisnosti koji na njoj vrijede: $F = \{ CD \rightarrow EF, DA \rightarrow B, H \rightarrow A, D \rightarrow GH \}$.
Ispitati vrijedi li funkcijska zavisnost $CD \rightarrow A$. Za svaki korak dokaza napisati pravilo koje se koristi.

8. (2 boda) Objasnite:

- zašto ne bi bilo dobro indeksirati svaki atribut relacije
- koje kriterije treba koristiti za odabir atributa nad kojima će se izgraditi indeks

9. (2 boda) Zadana je relacija STAZA prikazana na **Slici 1**. Napisati SQL naredbe koje će kreirati najmanji mogući broj indeksa za učinkovito obavljanje (pomoću B+ stabla) svih dolje navedenih upita.

- 1) `SELECT * FROM staza WHERE nazivStaza LIKE 'Sljeme%' AND tezina=5;`
- 2) `SELECT * FROM staza WHERE tezina >3 AND duljina<6;`
- 3) `SELECT * FROM staza WHERE datOdrzavanje BETWEEN '01.05.2016' AND '01.02.2017';`
- 4) `SELECT * FROM staza ORDER BY tezina, nazivStaza, datOdrzavanje;`
- 5) `SELECT * FROM staza ORDER BY tezina DESC, duljina DESC;`
- 6) `SELECT * FROM staza ORDER BY duljina, tezina;`

10. (3 boda) Relacija **MJESTO (pbr, naziv)** sadrži sljedećih 7 n-torki (10000, 'Zagreb'), (10410, 'Velika Gorica'), (10430, 'Samobor'), (21000, 'Split'), (31000, 'Osijek'), (51000, 'Rijeka'), (23000, 'Zadar').
Nacrtati dva B+ stabla reda 5 za atribut **pbr** 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 **pbr**.

Rješenja:

1. (3 boda)

```
SELECT sifStaza,
       nazivStaza,
       tezina,
       COUNT(sifObjekt) AS brojObjekata
FROM staza
NATURAL JOIN stazaObjekt
NATURAL JOIN PlDrustvo
WHERE nazivPlDrustvo LIKE '%HPD%'
GROUP BY sifStaza,
         nazivStaza,
         tezina
HAVING COUNT(sifObjekt) > 2;
```

2. (3 boda)

```
DELETE FROM objekt
WHERE sifTipObj IN (SELECT sifTipObj
                      FROM tipObjekta
                      WHERE LOWER(opis) LIKE '%utvrda%')
AND sifObjekt NOT IN (SELECT sifObjekt
                     FROM stazaObjekt);
```

alternativno rješenje

```
DELETE FROM objekt WHERE sifobjekt IN
(SELECT sifObjekt FROM objekt NATURAL JOIN tipobjekta
NATURAL LEFT JOIN stazaobjekt
WHERE sifstaza IS NULL
AND LOWER(opis) LIKE '%utvrda%');
```

još jedno rješenje:

```
DELETE FROM objekt
WHERE sifTipObj IN (SELECT sifTipObj
                      FROM tipObjekta
                      NATURAL JOIN objekt
                      WHERE LOWER(opis) LIKE '%ruše%')
AND sifObjekt NOT IN (SELECT sifObjekt
                     FROM stazaObjekt
                     NATURAL JOIN objekt);
```

3. (5 bodova)

```
SELECT nazivPlDrustvo,
       EXTRACT (YEAR FROM datOdrzavanje) AS godina,
       COUNT(DISTINCT staza.sifStaza) AS brRazlStaza
FROM Staza NATURAL JOIN StazaObjekt
RIGHT JOIN PlDrustvo ON Staza.sifPlDrustvo = PlDrustvo.sifPlDrustvo
AND EXTRACT (YEAR FROM datOdrzavanje)
> EXTRACT (YEAR FROM CURRENT_DATE) - 5
WHERE NOT(nazivPlDrustvo LIKE '%"%'
AND nazivPlDrustvo NOT LIKE '%"%"%')
-- moze i negacija uci u zgrade:
-- (nazivPlDrustvo NOT LIKE '%"%'
-- OR nazivPlDrustvo LIKE '%"%"%')
GROUP BY nazivPlDrustvo,
         godina
ORDER BY nazivPlDrustvo,
         godina DESC
```

4. (3 boda)

ρ REZ(drustvo, tezina, ukBroj, ukDuljina) (nazPIDrustvo, tezina \bowtie G COUNT(tezina), SUM(duljina) (PIDrustvo \bowtie ($\sigma_{tezina > 3}$ (Staza))))

5. (2 boda)

A

opis	brojObjekata
planinarski dom	2
bunar	1
ruševina	1
spomenik	0

6. (5 bodova)

1NF

IZLET = {OIB, siflzlet, datlzlet, imeP, prezP, pbrStan, nazivMjStan, trajanjelzlet, sifPlanina, nazivPlanina, sifVrh, nazivVrh, visinaVrh}

K = { OIB, siflzlet }

2NF

PLANINAR = {OIB, imeP, prezP, pbrStan, nazivMjStan}

IZLET1 = { siflzlet, datlzlet, trajanjelzlet, sifPlanina, nazivPlanina, sifVrh, nazivVrh, visinaVrh }

POHADJANJE = { OIB, siflzlet }

KPLANINAR = {OIB}

KIZLET1 = { siflzlet }

KPOHADJANJE = { OIB, siflzlet }

3NF

MJESTO = { pbr, nazivMj }

PLANINAR = {OIB, imeP, prezP, pbrStan }

VRH = { sifVrh, nazivVrh, visinaVrh, sifPlanina, nazivPlanina, }

IZLET2 = { siflzlet, datlzlet, trajanjelzlet, sifVrh }

PLANINA = { sifPlanina, nazivPlanina }

VRH1 = { sifVrh, nazivVrh, visinaVrh, sifPlanina }

POHADJANJE = { OIB, siflzlet }

KMJESTO = {pbr}

KPLANINAR = {OIB}

KVRH = { sifVrh } - NIJE U 3NF!

KIZLET2 = { siflzlet }

KPLANINA = {sifPlanina}

KVRH1= {sifVrh}

KPOHADJANJE = {OIB, siflzlet}

IZLET = {MJESTO, PLANINAR, IZLET2, PLANINA, VRH1, POHADJANJE}

7. (2 boda)

$CD \rightarrow CD$ (refleksivnost)

$CD \rightarrow CD$ i $CD \rightarrow EF \Rightarrow CD \rightarrow CDEF$ (akumulacija)

$CD \rightarrow CDEF$ i $D \rightarrow GH \Rightarrow CD \rightarrow CDEFGH$ (akumulacija)

$CD \rightarrow CDEFGH$ i $H \rightarrow A \Rightarrow CD \rightarrow CDEFGHA$ (akumulacija)

$CD \rightarrow CDEFGHA \Rightarrow CD \rightarrow A$ (dekompozicija)

8. (2 boda)

a)

- indeksi zauzimaju prostor
- operacija unosa ili brisanja n-torke koštaju tj. traju
 - uvijek rezultira promjenama (manjim ili većim) B-stabla
 - npr. ako je nad relacijom izgrađeno 10 različitih indeksa, unosom jedne n-torke u blokove s podacima morat će se unijeti zapisi i u 10 različitih indeksa
- operacija izmjene n-torke koštaju tj. traju
 - izmjena vrijednosti atributa A jedne n-torke rezultirat će brisanjem i dodavanjem zapisa u svim B-stablama za indekse u kojima se koristi atribut A

b)

- za attribute koji se često koriste za postavljanje uvjeta selekcije
- za attribute prema kojima se obavlja spajanje relacija
 - primarni i alternativni ključevi relacije
 - strani ključevi
- za attribute prema kojima se često obavlja sortiranje ili grupiranje

9. (2 boda)

```
CREATE INDEX i1 ON staza (tezina, nazivStaza, datOdrzavanje)  za upite 1 i 4
CREATE INDEX i2 ON staza (tezina, duljina)                  za upite 2 i 5
CREATE INDEX i3 ON staza (duljina, tezina)                  za upite 2 i 6
CREATE INDEX i4 ON staza (datOdrzavanje)                   za upit 3
```

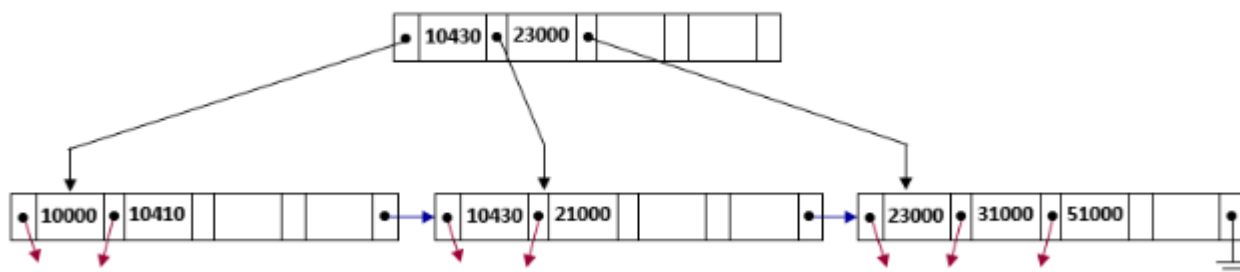
Može i ovako:

```
CREATE INDEX i1 ON staza (tezina DESC, nazivStaza DESC, datOdrzavanje DESC)
CREATE INDEX i2 ON staza (tezina DESC, duljina DESC)
CREATE INDEX i3 ON staza (duljina DESC, tezina DESC)
CREATE INDEX i4 ON staza (datOdrzavanje DESC)
```

10. (3 boda)

• Minimalno popunjeno stablo:

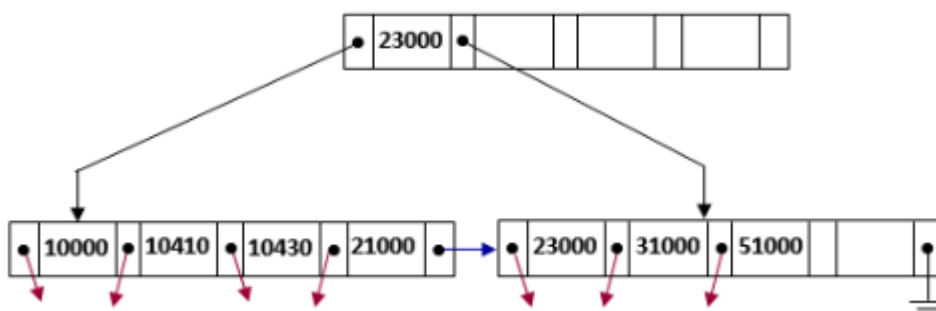
- list: minimalno 2
- interni čvor minimalno: 3, korijen 2



Ukupno **3 UI operacije**.

• Maksimalno popunjeno stablo:

- list: maksimalno 4
- interni čvor: maksimalno 5



Opet **3 UI operacije**!