

# Usmeni ispit iz Relacionih baza podataka, februar 2014. g. (I smer)

Broj indeksa	Ime i prezime

Zadaci se rade 180 minuta. PISATI ČITKO - NEČITKI ZADACI NEĆE BITI PREGLEDANI! Maksimalan broj poena je 80. Broj poena se izračunava tako što se saberu osvojeni poeni po zadacima i zbir podeli sa 2. Broj poena po zadacima je:

Zadatak	1	2	3	4	5	6	7	Zbir	Ukupno
<b>maks</b>	9	15	12	12	18	10	4	<b>80</b>	<b>Zbir/2</b>
<i>Osvojeno</i>									

1. a) Pokazati da je SQL relaciono kompletan, tj. da za proizvoljan izraz relacione algebre postoji semantički ekvivalentan SQL izraz. Pri tome, razmatrati samo izraze relacione algebre u kojima učestvuju osnovni Kodovi operatori.  
b) Šta je *Kodov algoritam redukcije*?
2. a) Formirati pogled *polaganje(rbr, ime, prezime, naziv smer, najviše\_puta\_polagan\_predmet, broj\_polaganja, poslednja\_ocena, prethodna\_ocena)* koji sadrži informacije o predmetu koga je student najviše puta polagao, koliko puta je polagao taj predmet, ocenu na poslednjem polaganju i ocenu na prethodnom polaganju. Ukoliko je neki predmet polagan najviše jedan put, tada za ocenu na poslednjem polaganju treba navesti '---'.  
b) Da li je preko ovog pogleda moguće ažurirati tabelu dosije? Ako jeste, uneti (preko pogleda) u tabelu dosije slog sa imenom i prezimenom studenta Petar Petrović i brojem indeksa 313/2013. Ukoliko nije, navesti zašto nije i uslove pod kojima u RSUBP DB2 može da se ažurira osnovna tabela preko pogleda koji je definisan nad njom.
3. Napišite upit kojim se prikazuju podaci o svim studentima koji su rođeni u mestu čiji naziv sadrži najviše dva blanko karaktera i sadrži 'nei' na 10 poziciji, i ne sadrži % niti \. Pri ispitivanju da li je na 10 poziciji niska 'nei' ne koristiti funkcije SUBSTR, SUBSTRING, SUBSTR2, SUBSTR4, SUBSTRB kao ni klauzulu LIKE. Naziv mesta rođenja je pisan bez blanko karaktera na početku ili kraju.
4. Neka je dat relvar  $R = \{A, B, C, D, E, F, G, H\}$  i skup F FZ:
  - 1)  $CD \rightarrow A$
  - 2)  $EC \rightarrow H$
  - 3)  $GHB \rightarrow AB$
  - 4)  $C \rightarrow D$
  - 5)  $EG \rightarrow A$
  - 6)  $H \rightarrow B$
  - 7)  $BE \rightarrow CD$
  - 8)  $EC \rightarrow B$

Odrediti sve kandidate za ključ relacije R. Obavezno obrazložiti sve korake u radu.

5. Neka je dat relvar  $R = \{B, C, D, F, G, H\}$  i skup FZ:

- 1)  $BG \longrightarrow CD$
- 2)  $G \longrightarrow F$
- 3)  $CD \longrightarrow GH$
- 4)  $C \longrightarrow FG$
- 5)  $F \longrightarrow D$

- a) Odrediti minimalni pokrivač skupa funkcionalnih zavisnosti
- b) Neka je relacija  $R_1 = \{C, D, G\}$  projekcija relacije  $R$ . Odrediti skup funkcionalnih zavisnosti koje su važeće u  $R_1$ .
- b) Dekomponovati relaciju  $R$  tako da novodobijeni skup relacija bude u BCNF.

Obavezno obrazložiti sve korake u radu.

6. a) Napisati SQL upit koji prikazuje komentare koji se odnose na tabele koje je napravio korisnik koji izvršava upit.
- b) Napisati SQL upit kojim se formira sinonim *Studenti* za tabelu *Dosije*
- v) Šta je efekat izvršavanja SQL upita

```
select ime, prezime, count(distinct id_smera)
from dosije
group by cube (ime, prezime);
```

Napisati **jedan** SQL upit, bez korišćenja GROUP BY CUBE ili drugih OLAP naredbi, koji proizvodi isti rezultat.

7. Formalno dokazati:

- a) Da je projekcija distributivna preko unije.
- b) Da je spajanje asocijativna operacija, ali da to nije i razlika.

**Uputstvo:** Napravite novi direktorijum i **obavezno** ga nazovite **Ime.Prezime.Januar.brojIndeksa**. Npr. ako student Pera Perić ima broj indeksa 125/2010, on treba da napravi direktorijum *Pera.Peric.Februar.2010.125*. U tom direktorijumu ostavite rešenja zadataka. Rešenje svakog zadatka ostavite u posebnoj datoteci. Datoteke sa rešenjima nazovite rednim brojem zadatka (npr. *zadatak1*, *zadatak2* ...). Za svaki zadatak ostavite tačno jedno rešenje, u suprotnom zadatak neće biti pregledan i biće ocenjen sa 0 (nula) poena.