

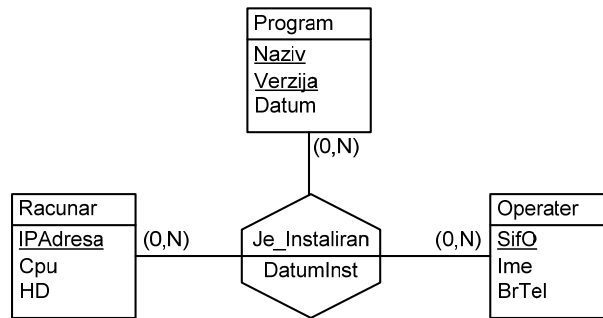
## Baze podataka

Ispit traje **180** minuta

Ime prezime i broj indeksa studenta	Potpis dežurnog	Broj poena

**Napomena:** Nije dozvoljena upotreba literature.

1. (6) Za model entiteta i odnosa, prikazan na slici, kreirati iskaz relacionog računa domena koji vraća nazive programa instaliranih na dva ili više računara.



Odgovor:

```

{< Naziv > | ∃ Verzija, IPAdresa1, SifO1, Dat1(
  < Naziv, Verzija, IPAdresa1, SifO1, Dat1 > ∈ Je_Instaliran ∧
  ∃ IPAdresa2, SifO2, Dat2 (< Naziv, Verzija, IPAdresa2, SifO2, Dat2 > ∈ Je_Instaliran ∧
  IPAdresa1 ≠ IPAdresa2))}
  
```

2.(4) Za model entiteta i odnosa iz prethodnog zadatka, kreirati SQL iskaz kojim se definiše ograničenje da sav softver na računaru mora biti instaliran od strane jednog operatera.

Odgovor:

```

CHECK ( NOT EXISTS ( SELECT J1.IPAdresa, J2.IPAdresa
                     FROM Je_Instaliran J1, Je_Instaliran J2
                     WHERE J1.IPAdresa = J2. IPAdresa
                           AND J1.SifO <> J2.SifO
                     )
)
  
```

3.(26) Dati su šema relacije  $R(A, B, C, D, E, F)$  i skup funkcijskih zavisnosti  $F=\{B \rightarrow CE, CF \rightarrow A, BD \rightarrow A, D \rightarrow E, ACF \rightarrow DE, B \rightarrow F\}$ . Potrebno je:

a) Odrediti skup kandidat ključeva  $KK$  date šeme.

---

Odgovor:  $KK=\{B\}$

---

b) Ispitati redom da li je data šema u BC, 3. i 2. normalnoj formi i svaki put u tabeli naznačiti da li posmatrana zavisnost narušava posmatranu normalnu formu.

	$B \rightarrow CE$	$CF \rightarrow A$	$BD \rightarrow A$	$D \rightarrow E$	$ACF \rightarrow DE$	$B \rightarrow F$
BCNF	√	×	√	×	×	√
3NF	√	×	√	×	×	√
2NF	√	√	√	√	√	√

c) Sprovesti normalizaciju date šeme u 3. normalnu formu algoritmom koji garantuje očuvanje funkcijskih zavisnosti.

---

Odgovor:

$R_1(B, C, F) R_2(C, F, A, D) R_3(D, E)$

---

d) Sprovesti normalizaciju date šeme u BC normalnu formu, izdvajajući zavisnosti redosledom sa desna na levo.

---

Odgovor:

$R_1(D, E) R_2(A, C, F, D) R_3(B, C, F)$

---

e) Ispitati da li je pri postupku u okviru tačke d) došlo do gubitka funkcijskih zavisnosti i kojih?

---

Odgovor:

Nema gubitka funkcijskih zavisnosti.

---

f) Napisati SQL skript koji na osnovu sadržaja tabele  $R$  popunjava tabele kreirane na osnovu rezultata dekompozicije dobijene u tački d). U svakoj od tabela naznačiti primarni ključ.

---

Odgovor:

$INSERT INTO R1(D, E) SELECT DISTINCT D, E FROM R;$   
 $INSERT INTO R2(A, C, F, D) SELECT DISTINCT A, C, F, D FROM R;$   
 $INSERT INTO R3(B, C, F) SELECT DISTINCT B, C, F FROM R;$

---

4.(34)

a) Ako se za oporavak od kvara koristi mehanizam Prateće stranice. Podaci A, B, C i D nalaze se u različitim stranicama na disku (podatak A u stranici 1,... , D u stranici 4). Prikazati izgled relevantnih delova sistema u trenutku kvara, neposredno posle trenutka  $t_{22}$ ? Koje operacije mehanizam Prateće Stranice vrši posle popravke sistema u cilju vraćanja baze u konzistentno stanje?

Odgovor:

T1	5	1	A	T1
	10	2	B	
		3	C	
		4	D	
		5	C1	
T2	7	6	C2	T2
	10	7	C3	
		8	B1	
		9	A1	
		10	D1	
T3	6	11		T3
	8	12		
	9	13		
		14		
		15		
		16		
		17		
		18		
		19		

Vreme	$T_1$	$T_2$	$T_3$
$t_1$	Read (C)		
$t_2$	$C := C + 1$		
$t_3$	Write (C)		
$t_4$			Read (C)
$t_5$			Read (B)
$t_6$			$C := C + B$
$t_7$			Write (C)
$t_8$			$B := B - 5$
$t_9$		Read (C)	
$t_{10}$		$C := 40$	
$t_{11}$		Write (C)	
$t_{12}$			Write (B)
$t_{13}$			Read (A)
$t_{14}$			$A := A + C$
$t_{15}$			Write (A)
$t_{16}$			Commit
$t_{17}$		Read (D)	
$t_{18}$		$D := D + C$	
$t_{19}$		Write (D)	
$t_{20}$		Commit	
$t_{21}$	Read (D)		
$t_{22}$	$D := D + C$		
$t_{23}$	Write (D)		
$t_{24}$	Commit		

Oporavak: **Restart{T<sub>1</sub>}**

b) Koje su vrednosti svakog od podatka u bazi nakon uspešnog izvršavanja posmatranog redosleda ukoliko podaci A, B, C i D na početku imaju vrednosti 10, 20, 30 i 40 respektivno?

Odgovor:

Podatak	Vrednosti
A	60
B	15
C	31
D	111

c) Da li je redosled sa slike serijalizabilan? Odrediti sve ekvivalentne serijske redoslede.

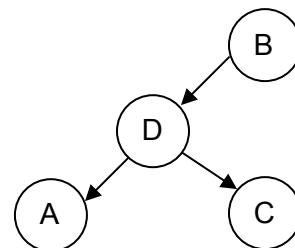
Odgovor: **Nije serijalizabilan.**

d) Da li je redosled prikazan na slici moguć ukoliko se uvede mehanizam zaključavanja po protokolu u obliku stabla? Ukoliko nije, navesti zašto nije i pokazati jedan mogući redosled izvršavanja, a ukoliko jeste onda treba popuniti datu tabelu za to izvršavanje (uz dodate instrukcije zaključavanja/otključavanja).

Odgovor: **Nije moguć (dati redosled nije serijalizabilan).**

Vreme	T <sub>1</sub>	T <sub>2</sub>	T <sub>3</sub>
t <sub>1</sub>	L(D)		
t <sub>2</sub>	L(C)		
t <sub>3</sub>	Rd(C)		
t <sub>4</sub>	Wr(C)		
t <sub>5</sub>	U(C)		
t <sub>6</sub>			L(B)
t <sub>7</sub>			L(D)
t <sub>8</sub>		L(D)	
t <sub>9</sub>	Rd(D)		
t <sub>10</sub>	Wr(D)		
t <sub>11</sub>	U(D)		
t <sub>12</sub>	Comm		
t <sub>13</sub>			L(D)
t <sub>14</sub>			L(C)
t <sub>15</sub>			Rd(C)
t <sub>16</sub>			Rd(B)
t <sub>17</sub>			Wr(C)
t <sub>18</sub>			U(C)
t <sub>19</sub>			W(B)
t <sub>20</sub>			U(B)
t <sub>21</sub>			L(A)
t <sub>22</sub>			U(D)
t <sub>23</sub>			Rd(A)
t <sub>24</sub>			Wr(A)
t <sub>25</sub>			U(A)
t <sub>26</sub>			Comm
t <sub>27</sub>		L(D)	
t <sub>28</sub>		L(C)	
t <sub>29</sub>		Rd(C)	
t <sub>30</sub>		Wr(C)	
t <sub>31</sub>		U(C)	
t <sub>32</sub>		Rd(D)	
t <sub>33</sub>		Wr(D)	
t <sub>34</sub>		U(D)	

t <sub>35</sub>		Comm	
t <sub>36</sub>			
t <sub>37</sub>			
t <sub>38</sub>			
t <sub>39</sub>			
t <sub>40</sub>			
t <sub>41</sub>			
t <sub>42</sub>			
t <sub>43</sub>			
t <sub>44</sub>			
t <sub>45</sub>			
t <sub>46</sub>			
t <sub>47</sub>			
t <sub>48</sub>			
t <sub>49</sub>			
t <sub>50</sub>			
t <sub>51</sub>			
t <sub>52</sub>			
t <sub>53</sub>			
t <sub>54</sub>			
t <sub>55</sub>			
t <sub>56</sub>			
t <sub>57</sub>			
t <sub>58</sub>			
t <sub>59</sub>			
t <sub>60</sub>			
t <sub>61</sub>			



e) Neka se za redosled iz tačke a) za oporavak od kvara koristi mehanizam Sistemskog Dnevnika sa Neodloženim Upisom. Prikazati izgled sistemskog dnevnika i navesti operacije koje sistem zadaje posle popravke sistema, u cilju vraćanja baze u konzistentno stanje ukoliko je kvar nastao neposredno posle trenutka  $t_{22}$ ?

Odgovor: <T1, start>

<T1, C, C0, C1>

<T3, start>

<T3, C, C1, C2>

<T2, start>

<T2, C, C2, C3>

<T3, B, B0, B1>

<T3, A, A0, A1>

<T3, commit>

<T2, D, D0, D1>

<T2, Commit>

Undo {T1}

Redo {T2, T3}

Restart {T1}

f) Koje su moguće vrednosti svakog od podataka u bazi neposredno pre trenutka  $t_{22}$  ako je izgled sistemskog dnevnika kao u tački pod e) i ukoliko pre početka izvršavanja transakcija podaci A, B, C i D imaju vrednosti 10, 20, 30 i 40 respektivno?

Odgovor:

Podatak	Moguće vrednosti
A	10, 61
B	20, 15
C	30, 31, 51, 40
D	40, 80

g) Koje operacije zadaje mehanizam u cilju vraćanja baze u konzistentno stanje u slučaju opisanom u tački pod e) ukoliko je poznato da je u trenutku neposredno posle Read(D) operacije transakcije  $T_2$  urađena kontrolna tačka koja je završena u trenutku neposredno posle Read(D) operacije transakcije  $T_1$ ? Kako će izgledati sistemski dnevnika u tom slučaju?

Odgovor:

<T1, start>

<T1, C, C0, C1>

<T3, start>

<T3, C, C1, C2>

<T2, start>

<T2, C, C2, C3>

<T3, B, B0, B1>

<T3, A, A0, A1>

<T3, commit>

<CHK (T1, T2)>

<T2, D, D0, D1>

<T2, Commit>

<CHK END>

Undo {T1}

Redo {T2}

Restart {T1}