40

מבחן במערכות בסיסי נתונים

רובי בוים

סמסטר א' מועד א' - 07 בפברואר, 2010
סמסטר א' מועד ב' - 20 בספטמבר, 2010
חומר סגור, שלוש שעות
סה"כ 100 נקודות

נא לכתוב את כל התשובות על הדפים
(המבחן כולל 13 דפים כולל דף זה)

מס מחברת: 40 מס סטודנט:

חלק	ציון
1	40
2	20
3	15
4	25
סה"כ	100

1. [35 נקודות]

כיוון שאתם סטודנטים מצטיינים, מארק צוקרברג החליט לאפשר לכם הצצה חלקית וחד פעמית לבסיס הנתונים של FACEBOOK:

Users(<u>uid</u>, name, birthday, country)

Events(<u>eid</u>, name, uid, date)

Guests (<u>eid</u>, <u>uid</u>, status)

לכל משתמש יש מזהה ייחודי, שם, תאריך לידה ומדינה. עבור כל אירוע המערכת שומרת את מזהה ייחודי ובנוסף את שם האירוע, המשתמש שערך את האירוע והתאריך. בנוסף, אירועים יכולים להכיל רשימת מוזמנים שלכל אחד סטטוס הגעה (yes/no/maybe).

א. מארק (uid=123) רוצה לברר למה צלמי הפפראצי לא מגיעים לאירועים שהוא עורך. עזרו לו לכתוב שאילתה שתחזיר את רשימת כל האנשים (<u>מזהה ושם</u>) <u>שערכו</u> אירוע <u>בזמן מקביל</u>.

```
SELECT DISTINCT U.uid , U.name
FROM Users U , Events E1 , Events E2
WHERE E1.uid = '123' AND E2.uid <> '123' AND
      E1.date = E2.date AND
      E2.uid = U.uid
```

✓

ב. כתבו את אותה שאילתה באלגברה רלציונית.

$$\Pi_{uid,name}\Big[\Big[\rho_{E1.uid,\,date}\big(\Pi_{uid,date}\big(\sigma_{uid='123'}(Events)\big)\big) \bowtie_{date=date}$$

$$\Pi_{uid,date}\big(\sigma_{uid<>'123'}(Events)\big)\Big] \bowtie_{uid=uid}$$

$$\big[\Pi_{uid,name}(Users)\big]\Big]$$

ג. בעקבות הצלחתכם (בתקווה ☺) בסעיפים הקודמים, מארק מתעניין בהשקעה חדשה בשוק הישראלי. אך הוא עדיין מודאג שFACEBOOK אינה מספיק פופולארית בקרב המשתמשים המקומיים. עזרו לו לכתוב שאילתה שתחזיר את מספר האירועים שנערכו על ידי משתמשים ישראלים אשר הכילו לפחות 100 אנשים (כלשהם) שהסטטוס שלהם הוא yes או maybe.

```
SELECT  COUNT ( E.eid)
FROM   Events E , Users U
WHERE  U.country = 'Israel' AND U.uid = E.uid AND
       E.eid  IN  (SELECT  E2.eid
                   FROM  Events E2 , Guests G
                   WHERE  E2.eid = G.eid  AND
                          (G.status = 'yes'  OR
                           G.status = 'maybe')
                   GROUP BY  E2.eid
                   HAVING  COUNT(G.*) >= 100)
```

ד. התבוננו בזוגות Q1, Q2 של השאילתות הבאות. לכל זוג סמנו מי מהמשפטים הבאים מתקיים:

(A) שתי השאילתות מחזירות תמיד בדיוק אותה תשובה

(B) התשובות המחוזרות ע"י Q1 מוחזרות גם ע"י Q2, אבל יתכן ש Q2 תחזיר ערכים נוספים או כפילויות של ערכים.

(C) התשובות המחוזרות ע"י Q2 מוחזרות גם ע"י Q1, אבל יתכן ש Q1 תחזיר ערכים נוספים או כפילויות של ערכים.

(D) אף אחד מהנ"ל

דוגמה 1:

Q1 = **SELECT** * **FROM** events **WHERE** date > 01-01-2010

Q2 = **SELECT** * **FROM** events **WHERE** date > 01-01-2009

עליכם לענות (B)

דוגמה 2:

Q1 = **SELECT** name **FROM** Guests

Q2 = **SELECT DISTINCT** name **FROM** Guests

עליכם לענות (C)

i.

```
Q1 =  SELECT  eid
      FROM    Events
```

```
Q2 =  SELECT  eid
      FROM    Guests
```

הערה: אפשר להניח שמוגדר foreign key בטבלת Guests על שדה eid

(A) (B) (C) (D)

ii.

```
Q1 =  SELECT  U.uid
      FROM    Users U
      WHERE   1     >=
              (     SELECT    count(*)
                    FROM      Events E, Guests G
                    WHERE     E.eid   =   G.eid   AND
                              E.uid   =   U.uid   AND
                              G.uid   =   U.uid   )
```

```
Q2 =  SELECT  E.uid
      FROM    Events E, Guests G
      WHERE   E.uid   =   G.uid   AND
              E.eid   =   G.eid
```

(A) (B) (C) (D)

iii.

Q1 = **SELECT** **DISTINCT** U.uid, U.name
FROM Users U, Guests G1, Guest G2
WHERE G1.uid = 123 AND
G1.eid = G2.eid AND
G2.uid = U.uid

Q2 = **SELECT** uid, name
FROM Users
WHERE uid **IN**
(SELECT uid FROM Guests WHERE eid IN
(SELECT eid FROM Guests WHERE uid = 123)

(A) (B) (C) (D)

2. [20 נקודות]

נתונה הטבלה **R(A, B, C, D, E)** עם התלויות הפונקציונאליות הבאות:

A → C

B, C → E

B → D

א. מצא את המפתח (מינימאלי) של טבלה R

המפתח המינימלי הוא {A, B}

ב. פרקו את הטבלה לטבלאות BCNF ונמקו את תשובתכם. התשובה צריכה להכיל:

- רשימת טבלאות והשדות בכל טבלה
- המפתחות בכל אחת מהטבלאות

תלויות נוספות:
B, C → D, E

R(A, B, C, D, E)

B, C → D, E — הפרה את BCNF כי B, C לא מפתח

R1(B, C, D, E) — B → D הפרה את BCNF כי B לא מפתח

R2(A, B, C) — A → C הפרה את BCNF כי A לא מפתח

R1(B, D) — B → D

R2(B, C, E) — B, C → E

R3(A, C) — A → C

R4(A, B) — ריק FDs

הרשימה הסופית: R1(B, D) עם המפתח {B}, R2(B, C, E) עם המפתח {B, C}, R3(A, C) עם המפתח {A} ו- R4(A, B) עם המפתח {A, B} מכיוון שאלו הטבלאות שמכילות את כל ה-FD's שמפרות את BCNF, ועדיין משמרות את כל ה-FD's המקוריים (ה-non-trivial) שיש ביניהם (ולא נוצרים FD's חדשים שלא היו קיימים קודם).

ג. עבור כל טבלה שקיבלתם (בפרוק הסופי) ציינו האם היא במקרה גם ב 3NF

ריק כי BCNF ⊂ 3NF וגם כל טבלה שקיבלנו בפירוק הסופי, שהיא טבלה ב- BCNF, היא גם טבלה ב- 3NF.

3. [20 נקודות]

נתון מסמך ה XML הבא המתאר חנות משחקי ווידאו. המוצרים נחלקים לפי סוג קונסולת המשחק (XBOX, WII, PLAYSTATION..) אשר לכל אחת יש שם, יצרן ורשימת משחקים. עבור כל משחק שומרים את השם ואת מספר המשתתפים המקסימלי האפשרי.

Input.xml

```
<rubi-store>
    <console>
        <name>Xbox360</name>
        <manufacture>Microsoft</manufacture>
        <games>
            <game players=1>halo</game>
            <game players=4>guitar hero 5</game>
            ...
        </games>
    </console>
    <console>
        <name>Wii</name>
        ...
    </console>
    ...
</rubi-store>
```

א. כתוב ביטוי XPath שמחזיר את כל המשחקים בעלי אפשרות משחק ל4 שחקנים בדיוק עבור קונסולת ה Xbox360. התשובה צריכה להראות בסגנון הבא:

guitar hero 5
call of duty 4
...

```
//console [/name/text() = 'Xbox360'] /games/game [@players = '4']/text()
```

✓

ב. כתוב ביטוי XQuery שמחזיר את רשימת הקונסולת ואת כמות המשחקים המתאימים שלהם. סננו קונסולות עם פחות מ 50 משחקים. התשובה צריכה להראות בסגנון הבא:

```
<result>
    <ConsoleName>Xbox360</ConsoleName >
    <ConsoleCount>322</ConsoleCount>
</result>
<result>
    <ConsoleName>Wii</ConsoleName >
    <ConsoleCount>65</ConsoleCount>
</result>
...
```

✓

(הערה: ודאו כשלוש קונסולות ואחת לפחות אין שלושים משחקים.)

```
FOR $c IN document('Input.xml')//console
LET $g := count(document('Input.xml')//console[/name/text() = $c/name/text()]
          /games/game)
WHERE $g >= 50
RETURN { <result>
           <ConsoleName> $c/name/text() </ConsoleName>
           <ConsoleCount> $g </Console Count>
         </result> }
```

4. [25 נקודות]

נתונות שתי הטבלאות R(A, B, C) S(D, E, F) עם האינדקסים הבאים:

א. על השדה R.A (מסוג unclustered)

ב. על השדה R.B (מסוג clustered)

ג. על השדה S.D (מסוג clustered)

נניח ששתי הטבלאות גדולות ולא ניתן להחזיק אותן בזיכרון, אבל האינדקסים קטנים וכן יושבים בזיכרון בשלמותם.

א. נתבונן בשאילתה הבאה:

SELECT * **FROM** R,S **WHERE** R.C = S.D

נניח שמימוש ה JOIN בשאילתה הוא ע"י אלגוריתם Index nested join.

האם ישפיע שינוי האינדקס על שדה S.D ל unclustered על ביצועי המערכת בחישוב השאילתה?

(נמקו בקצרה – שתי שורות לכל היותר!)

השינוי לא ישפיע כיוון ש-S הוא מפתח ב-S, ולכן בכל מקרה יגשו לכל היותר אחד מהרשומות לפי הקריטריון (כיוון שגם מקרה האינדקס בזיכרון, אלא 0/3 על גישה לדיסק הוא 0).

ב. נתון המידע הסטטיסטי הבא:

B(R) = 100	(מספר הדפים בטבלה R)
T(R) = 2000	(מספר הרשומות בטבלה R)
V(R,B) = 20	(מספר הערכים השונים בשדה B בטבלה R)
B(S) = 20	(מספר הדפים בטבלה S)
T(S) = 1000	(מספר הרשומות בטבלה S)
M=80	(גודל הזכרון בדפים)

נתבונן בשאילתה הבאה:

SELECT * FROM R,S **WHERE** R.C = S.D **AND** R.B = 123

עבורה, נתונות תוכניות הביצוע הלוגיות הבאות:

$P1 = S\ join_{D=C}\ (select_{B=123}(R))$

$P2 = select_{B=123}(S\ join_{D=C}\ R)$

נניח שכל אחת מתוכנית הביצוע {P1, P2} מחושבת ב pipeline, כלומר תוצאת אופרטור אחד ניתנת ישירות לאופרטור הבא (בלי להיכתב לדיסק בדרך).

יש שני אופרטורים בסיסים:

- JOIN: נניח שממומש על ידי ~~Sort-Merge~~ hash Join
- SELECT: יכול להיות ממומש על ידי האינדקס או ע"י מעבר סידרתי על הטבלה. כאשר הוא לא האופרטור הראשון, הוא ניתן לביצוע ישירות על ה tuple-ים שהוא מקבל כקלט (בלי תשלום של I/O).

חשבו את מחיר כל אחת מתוכנית הביצוע במונחים של פעולות I/O (לא כולל כתיבת התוצאה הסופית לדיסק). בטאו כל אחת מהתשובות כפונקציה של הסטטיסטיקות הנ"ל. לדוגמה:

COST(P1) = B(R)B(S)/ V(R,A) (<u>לא התשובה האמיתית!</u>)

הסבירו כל תשובה בקצרה (2 שורות מקסימום)

A. COST(P1, ע"י אינדקס select) = $\lceil \frac{B(R)}{V(R,B)} \rceil + 3 \cdot B(S) + 2 \cdot \lceil \frac{B(R)}{V(R,B)} \rceil =$

$= \frac{100}{20} + 3 \cdot 20 + 2 \cdot \frac{100}{20} = 5 + 60 + 10 = 75$ ✓

הואיל והתנאי הוא על אחד האטריבוטים שעליהם יש אינדקס clustered, ושני הטבלאות הן ממוינות לפי אותם האטריבוטים, אז אופרטור ה- hash join, יעשה רק מעבר אחד בזמן שנקרא/נכתב ל- R ויהיה לכן $\lceil \frac{B(R)}{V(R,B)} \rceil$.

B. COST(P1, select ע"י מעבר סידרתי) = $B(R) + 3B(S) + 2 \cdot \frac{\lceil B(R) \rceil}{V(R,B)}$ =

= $100 + 3 \cdot 20 + 2 \cdot \frac{100}{20}$ = $100 + 60 + 10$ = 170

הסיבה העיקרית היא שלא צריך לבצע select - מאחר ואין צורך לסרוק את כל הקובץ R. אנחנו יודעים מה שלוח ה-hash, ומאחר שכבר R ממוין לפי B(R)/V(R,B) אנחנו יודעים שאחרי ה-select אותם

C. COST(P2, select ע"י אינדקס) = $3 \cdot B(S) + 3 \cdot B(R) = 3 \cdot 20 + 3 \cdot 100 = 360$

הסיבה העיקרית היא שלא צריך לבצע select - מאחר ואין צורך לסרוק את כל הקובץ R. אנחנו יודעים מה שלוח ה-hash, ומאחר שכבר R ממוין לפי B(R)/V(R,B) אנחנו יודעים שאחרי ה-select אותם

D. COST(P2, select ע"י מעבר סידרתי) = $3 \cdot B(S) + 3 \cdot B(R)$ = 360

הסיבה היא אותה סיבה, שוב, שלא צריך לבצע select - מאחר ואין צורך לסרוק את כל הקובץ R. אנחנו יודעים מה שלוח ה-hash ומאחר שכבר ממוין

ג. ציינו מהי התוכנית הזולה ביותר לחישוב השאילתה, ואת מחירה (מבוטא כמספר)

לפי הנתונים שהיו לנו, הזולה ביותר היא התוכנית P1 שכן עלותה הכוללת היא 75 פעולות I/O.