

מבחן במערכות בסיסי נתונים

רובי בויס

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: _____ מס סטודנט: _____

חלק	ציון
1	
2	
3	
4	
סה"כ	

1. [35 נקודות]

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

Users(uid, name, birthday, country)

Events(eid, name, uid, date)

Guests (eid, uid, 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 (מזהה ושם) שערכו אירוע בזמן מקביל.

ב. כתבו את אותה שאילתה באלגברה רלציונית.

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 שאלתה שתחזיר את מספר האירועים שנערכו על ידי משתמשים ישראלים אשר הכילו לפחות 100 אנשים (כלשהם) שהסטטוס שלהם הוא yes או maybe.

ד. התבוננו בזוגות Q1, Q2 של השאלות הבאות. לכל זוג סמנו מי

מהמשפטים הבאים מתקיים:

- (A) שתי השאלות 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 הפונקציונליות הבאות:

$$A \rightarrow C$$

$$B, C \rightarrow E$$

$$B \rightarrow D$$

א. מצא את המפתח (מינימאלי) של טבלה R

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

- רשימת טבלאות והשדות בכל טבלה
- המפתחות בכל אחת מהטבלאות

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

ב. כתוב ביטוי XQuery שמחזיר את רשימת הקונסולות 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>  
...
```

4. [25 נקודות]

נתונות שתי הטבלאות $R(\underline{A}, B, C)$ $S(\underline{D}, E, F)$ 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!)

ב. נתון המידע הסטטיסטי הבא:

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

נתבונן בשאילתה הבאה:

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

עבורה, נתונות תוכניות הביצוע הלוגיות הבאות:

$P1 = S \text{ join}_{D=C} (\text{select}_{B=123}(R))$

$P2 = \text{select}_{B=123}(S \text{ join}_{D=C} R)$

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 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 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:

$COST(P1) = B(R)B(S)/V(R,A)$ (לא התשובה האמיתית!)
הסבירו כל תשובה בקצרה (2 שורות מקסימום)

A. $COST(P1, \text{select ע"י אינדקס}) =$

B. $\text{COST}(P1, \text{select ע"י מעבר סידרתי}) =$

C. $\text{COST}(P2, \text{select ע"י אינדקס}) =$

D. $\text{COST}(P2, \text{select ע"י מעבר סידרתי}) =$

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