

סמסטר חורף התשפ"א

מרצה: פרופ' חגית עטיה

מתרגלים: אסף ישורון

אלעזר גרשוני

נועה שילר

גל גרימברג

מסדי נתונים

236363

מועד א'

15 בפברואר 2021

פירוט החלקים והניקוד:

שאלה	נושא	ניקוד	הערות
1	SQL	20	
2	RA	20	
3	Design	20	
4	XML	20	
5	Concurrency Control	20	

הנחיות לנבחנים

- יש להשתמש רק בסימנים או פונקציות שנלמדו בתרגול או בהרצאה ומופיעים בשקפים של הקורס. כל שימוש בסימון שאינו כזה מחייב הסבר מלא של משמעות הסימון.
- הזמן המוקצה למבחן הינו שלוש שעות, תכננו את הזמן בהתאם.
- אין לכתוב בעפרון.
- ניתן להשתמש בכל חומר עזר שנגיש ישירות מה-Moodle. אין להשתמש במקלדת.

בהצלחה!

:SQL

בחנות הפרחים Bouequet2U מוצעים למכירה זרי פרחים שונים. נתון מסד הנתונים הבא, המכיל את הרכבי הזרים המוצעים למכירה ואת ההזמנות המגיעות לחנות:

Flowers(fID, fName, color, price)

הטבלה מכילה את הפרחים השונים מהם ניתן להרכיב זרים. לכל פרח שמור שמו, צבע הפרח ומחירו, וכל פרח כזה מזוהה ע"י מזהה יחודי fID. ניתן להניח כי שילוב של שם פרח וצבע מסויים ימצא לכל היותר פעם אחת בטבלה, כלומר שהטבלה מקיימת את המגבלה UNIQUE(fName,color).

Bouquets(bID, fID, quantity)

הטבלה מכילה את הזרים השונים המוצעים למכירה ואת הרכבם. השדה bID מכיל את המזהה של הזר, וכל שורה מציינת סוג פרח וכמות הפרחים מסוג זה הנמצאים בזר. שימו לב כי זר פרחים עם מזהה מסויים מוגדר ע"י כל השורות הנמצאות בטבלה זו בהן המזהה מופיע.

Orders(ordID, bID, quantity)

הטבלה מכילה את ההזמנות המגיעות לחנות. המזהה ordID מכיל את המזהה של ההזמנה וכל שורה מכילה זר והכמות ממנו הכלולים בהזמנה. שימו לב כי הזמנה עם מזהה מסויים מוגדרת ע"י כל השורות הנמצאות בטבלה זו בהן המזהה מופיע.

שימו לב: מפתחות הסכמות מסומנים בקו תחתון.

דוגמה למסד נתונים אפשרי:

Flowers			
<u>fID</u>	fName	color	price
1	"Fern"	"green"	2
2	"Sunflower"	"yellow"	10

Bouquets		
<u>bID</u>	<u>fID</u>	quantity
1	1	5
1	2	10
2	2	10

Orders		
<u>ordID</u>	<u>bID</u>	quantity
1	1	2
1	2	3

ענו על הסעיפים הבאים. ניתן להשתמש בשאילתות מקוננות ובמבטים (VIEWS).

א. (4 נקודות) כתבו שאילתת SQL המחזירה את כל הזוגות (bID, color), כך שבזר עם המזהה bID קיים פרח בצבע color. על כל זוג להופיע לכל היותר פעם אחת בתוצאה של השאילתה.

```
SELECT DISTINCT bID, color
FROM Bouquets B, Flowers F
WHERE B.fID = F.fID;
```

עוד אפשרות המבטיחה את הייחודיות של כל זוג בתוצאה היא להשתמש ב- GROUP BY bid, color.

ב. (4 נקודות) כתבו שאילתת SQL המחזירה עבור כל הזמנה ומזהה של פרח הכלול בהזמנה את כמות הפרחים מסוג זה הכלולים בהזמנה (כלומר, הסכום על פני כל הזרים בהזמנה). אם הפרח לא מופיע בהזמנה, אין צורך להחזיר שורה עבור צירוף זה. לדוגמה, עבור מסד הנתונים הנתון יוחזרו כל השלשות הבאות:

ordID	fID	Quantity
1	1	$2 \times 5 = 10$
1	2	$2 \times 10 + 3 \times 10 = 50$

```
SELECT OrdID, fID, SUM(O.quantity * B.quantity)
FROM Orders INNER JOIN Bouquets B
ON O.bid = B.bid
GROUP BY OrdID, fID;
```

מחיר של זר מוגדר להיות סכום המחירים של כל הפרחים הכלולים בזר. לדוגמה, עבור מסד הנתונים הנתון מחיר הזר עם המזהה 1 הוא: $5 \times 2 + 10 \times 10 = 110$.

ג. (4 נקודות) כתבו שאילתת SQL המחזירה את כל הזוגות (bid, price) כך ש-bid הוא מזהה של זר ו-price הינו המחיר הכולל של הזר.

```
SELECT bid, SUM(B.quantity * F.price) as price
FROM Bouquets as B INNER JOIN Flowers as F
ON F.fID = B.fID
GROUP BY bid;
```

הניחו כי תוצאת השאילתה מהסעיף הקודם נמצאת בטבלה Prices(bid,price). ניתן להשתמש בטבלה זו בסעיף הבא, גם אם לא פתרתם את הסעיף הקודם.

ד. (8 נקודות) כתבו שאילתת SQL המחזירה את מזהי כל ההזמנות המקיימות לפחות אחד משני התנאים הבאים:

1. ההזמנה כוללת זר שמחירו גדול או שווה למחיר הממוצע עבור זרים המכילים יותר מ-10 פרחים (אם אין זרים עם יותר מ-10 פרחים, המחיר הממוצע הוא 0).
2. ההזמנה כוללת זר המכיל פרח בצבע אדום (כלומר, Flowers.color = 'red').

```
SELECT ordID
FROM Orders O, Prices P
WHERE O.bID = P.bID AND
P.price >= (SELECT COALESCE(AVG(price),0)
           FROM Prices
           WHERE Prices.bID IN (SELECT bID
                                FROM Bouquets B
                                GROUP BY bID
                                HAVING SUM(quantity) > 10))

UNION

SELECT ordID
FROM Orders
WHERE bID IN (SELECT bID
              FROM Bouquets B, Flowers F
              WHERE B.fID = F.fID AND F.color = 'red');
```

:RA

נתונות שתי רלציות: Vertex ו-Edges המייצגות ביחד גרף מכוון G , כך שכל רשומה ב-Vertex מייצגת צומת ב- G וכל רשומה ב-Edges מייצגת קשת מכוונת ב- G . ניתן להניח שהרלציה Edges מכילה רק צמתים שנמצאים גם ברלציה Vertex. בנוסף ניתן להניח שאין בגרף קשתות עצמיות.

דוגמה להמחשה:

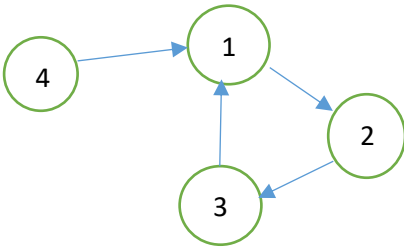
Edges

V ₁	V ₂
1	2
2	3
3	1
4	1

Vertex

V ₁
1
2
3
4

גרף G המתאים לרלציות:



א. כתבו שאילתת RA שמחזירה עמודה אחת המכילה את כל הצמתים שנמצאים במעגל בעל 3 קשתות בדיוק ב- G . (4 נק')

$$\pi_{v1} \left(\sigma_{v_3=v'_3 \wedge v_1=v'_1} \left(\sigma_{v_2=v'_2} \left(Edges \times \rho_{v1 \rightarrow v'_2, v2 \rightarrow v3} Edges \right) \right) \times \rho_{v1 \rightarrow v'_3, v2 \rightarrow v1'} Edges \right)$$

ב. פתרו שוב את סעיף א', הפעם על ידי שאילתת RC. (4 נק')

$$\{v \mid Vertex(v) \wedge \exists v_1, v_2 \ Edges(v, v_1) \wedge Edges(v_1, v_2) \wedge Edges(v_2, v)\}$$

ג. נתונה שאילתת RC הבאה:

$$Q = \{v \mid \exists v_1 \ Edges(v, v_1) \wedge \exists v_2 \neg Edges(v_2, v)\}$$

הסבירו מדוע Q היא תלויה בתחום (Domain Dependant). (6 נק')

נסביר על ידי הדוגמא הבאה:

$$Edges = \{(1,1)\}$$

$$D1 = \{1\} \rightarrow output : Empty$$

$$D2 = \{1,2\} \rightarrow output : \{1\}$$

ד. כתבו שאילתת RA המחזירה את כל הצמתים שניתן להגיע מהם לכל שאר צמתי הגרף (למעט עצמם) בקשת אחת ישירה. (6 נק')

$$G' = Edges \cup \sigma_{v_2=v_1} (\rho_{v_1 \rightarrow v_2} Vertex \times Vertex)$$

$$Answer = G' \div Vertex$$

Design:

א. עבור הסכמה $R = \{A, B, C, D, E\}$ נגדיר:

$$F = \{A \rightarrow B, \quad B \rightarrow C, \quad AB \rightarrow C, \quad D \rightarrow ED, \quad E \rightarrow ED\}$$

1. מצאו כיסוי מינימלי עבור (R, F) . אין צורך להסביר את תשובתכם. (3 נק')

$$F' = \{A \rightarrow B, \quad B \rightarrow C, \quad D \rightarrow E, \quad E \rightarrow D\}$$

2. מצאו מפתח (קביל, מינימלי) עבור הסכמה. הסבירו. (3 נק')

מפתח קביל לדוגמה: AD . מתקיים $A_F^+ = ABC$ וגם $D_F^+ = DE$ ולכן $AD_F^+ = R$. בנוסף A בלבד או D בלבד לא יספיקו.

3. מצאו פירוק 3NF משמר מידע ותלויות עבור הסכמה. אין צורך להסביר את תשובתכם. (3 נק')

$$D = \{AB, BC, DE, AD\}$$

לפי האלגוריתם מההרצאה:

ב. עבור הסכמה $R = \{A_1, A_2, \dots, A_k\}$, נגדיר את קבוצת התלויות הבאה:
 $G = \{X \rightarrow Y \mid X \subseteq R, Y \subseteq R\}$

הערה חשובה: כוונת המשורר בשאלה הזאת הייתה לא לקחת בחשבון את הקבוצה הריקה בתלויות של G . אם כן לוקחים את הקבוצה הריקה בחשבון – השאלה הופכת לקלה בהרבה. כיוון שה-"פרצה" הזאת בשאלה התגלתה בשלב מאוחר יחסית במבחן, החלטנו לא להעביר תיקון גורף על מנת שלא לפגוע בסטודנטים שכבר פתרו את השאלה על פי פרשנות מסוימת, אלא פשוט לקבל את שתי הפרשנויות (בהנחה שהפתרונות נכונים כמובן). בשורה התחתונה, מדובר היה בסטודנטים בודדים שפתרו את השאלה ולקחו בחשבון את הקבוצה הריקה, והרוב פתרו על פי הכוונה המקורית של השאלה.

1. מצאו כיסוי מינימלי עבור (R, G) , הסבירו. (3 נק')

לדוגמה: $G' = \{A_1 \rightarrow A_2, A_2 \rightarrow A_3, \dots, A_k \rightarrow A_1\}$. ייתכנו עוד k כיסויים מינימליים נכונים k "כוכבים" שבמרכזם אחד האטריביוטים, ועוד אפשרויות בנוסף לזה.

2. תהי r רלציה בעלת 6 רשומות שונות המתאימה לסכמה R ומקיימת את כל התלויות ב- G . כמה רשומות שונות ייתכנו ברלציה $\pi_{A_1}(r)$? מהן האפשרויות? הסבירו. (4 נק')

ייתכנו 6 רשומות בלבד. ראשית, כמובן שלא ייתכנו יותר מ-6 רשומות. שנית, 6 רשומות ייתכנו כאשר כל שתי רשומות ב- r שונות לחלוטין אחת מהשנייה (כל שתי רשומות נבדלות בכל אחד מהאטריביוטים). כעת נניח בשלילה שיש פחות מ-6 רשומות. מכאן שיש שתי רשומות ב- r שהן שונות (נתון שכל הרשומות ב- r שונות) אך בעלות אותו ערך עבור A_1 . נניח ששתי הרשומות נבדלות בערך האטריביוט A_i . במקרה כזה r לא מספקת את $A_1 \rightarrow A_i$ בסתירה לכך ש- r מספקת את G .

3. האם ייתכן כיסוי מינימלי עם פחות מ- k תלויות? הסבירו. (4 נק')

לא ייתכן. נניח בשלילה שקיים כזה G' . בכיסוי מינימלי מופיע באגף ימין של כל תלות אטריביוט אחד בלבד. לכן, כיוון שב- G' קיימות פחות מ- k תלות - קיים אטריביוט A_i שלא מופיע באגף ימין של אף תלות ב- G' . עבור אטריביוט כלשהו אחר A_j מתקיים $A_i \in A_j^+$, אבל $A_i \notin A_j^+$, סתירה.

:XML

שאלה זו תעסוק בטבלאות html, אליהן נתייחס כמסמכי XML. נעסוק בטבלאות המכילות עמודות עבור שם פרטי, שם משפחה וגיל, כפי שמראה הדוגמה הבאה:

```
<table>
  <tr><th>First</th><th>Last  </th><th>Age</th></tr>
  <tr><td>Jill </td><td>Smith  </td><td>50 </td></tr>
  <tr><td>Eve </td><td>Jackson</td><td>24 </td></tr>
  <tr><td>Jane </td><td>Jane  </td><td>100</td></tr>
</table>
```

הניחו שבטבלאות אין תגים מלבד אלה שמופיעים בדוגמה, ושמבנה הטבלה תקין.

א. כתבו שאילתת X-Path שמחזירה את העמודה השלישית בטבלה, כולל כותרת (<th> ורשימה של <td>). (4 נק')

`//tr/(th|td)[3]`

ב. כתבו שאילתת X-Path שמחזירה את צמתי הגילאים של אנשים ששמן הפרטי זהה לשם המשפחה. (4 נק')

`//tr[td[1]=td[2]]/td[3]`

ג. סדר העמודות בטבלה השתבש (בסעיף זה בלבד) באופן לא ידוע. כתבו ביטוי שמחזיר עמודה בטבלה (רשימה של <td>) בהינתן שהטקסט בכותרת העמודה (<th>) הוא "First". (4 נק')

`//td[count(//th[text()='First']/preceding-sibling::th)+1]`

ד. כתבו ביטוי שמחזיר את השורות בהן הגיל גדול או שווה לשורה הקודמת. אין צורך להחזיר את השורה הראשונה, וניתן להניח שהגיל שמופיע בה הוא 0. הערה:

לשם התייחסות לטקסט כמספר ולא כמחרוזת, יש להשתמש ב `number()` ולא `text()`. (4 נק')

`//tr[td[3]/number()>=preceding::tr/td[3]/number()]`

ה. נתון שהטבלה עצמה (ללא הכותרות) היא ריבועית (מספר השורות שווה למספר העמודות). כתבו ביטוי אשר מחזיר את האלכסון הראשי בטבלה: תא ראשון בעמודה ראשונה, תא שני בעמודה שנייה, וכן הלאה. עבור הטבלה שבדוגמה, התוצאה צריכה להיות (4 נק')

`<td>Jill</td><td>Jackson</td><td>100</td>`

`//td[count(..preceding-sibling::tr)]`

Concurrency Control

נתונות n טרנסאקציות T_1, T_2, \dots, T_n , ושני תזמונים (schedules) S_1, S_2 של הטרנסאקציות הללו.

א. עבור כל אחד מהמקרים הבאים ציינו האם ניתן לקבוע כי S_1 ו- S_2 הם שקולי קונפליקט. (בוודאי שקולים, בוודאי לא שקולים, אולי כן ואולי לא). נמקו בקצרה (משפט אחד).

א. (3 נקודות) גרף הקונפליקט של S_1 זהה לגרף הקונפליקט של S_2 ואין בו מעגלים. בוודאי שקולים, כי שניהם שקולים לתיזמון סדרתי.

ב. (4 נקודות) גרף הקונפליקט של S_1 זהה לגרף הקונפליקט של S_2 ויש בו מעגלים.

בהגדרות מסוימות של גרף הקונפליקט (precedence graph), קשתות בגרף הקונפליקט מסומנות בשם האובייקט שעליו הקונפליקט. תחת הגדרה כזאת, התשובה הנכונה לשאלה היא: תמיד שקולים, כי גרף הקונפליקט מקדיש לכל קונפליקט קשת ומכוון אותה מהטרנסאקציה המוקדמת בתזמון לטרנסאקציה המאוחרת בתזמון. כך שגרף קונפליקט זהה אומר שסדר הקונפליקטים (על אותו אובייקט) זהה.

עם זאת, ההגדרה שניתנה בקורס לא מסמנת את קשתות הקונפליקט בשמות אובייקטים, ולכן יכול להיות מעגלים זהים בגרפים של S_1 ו- S_2 (אשר מסומנים באופן הפוך) ושני התזמונים אינם שקולי קונפליקט.

כמובן, ייתכן גם שהם שקולי קונפליקט.

בגלל הבלבול, הסעיף נבדק מחדש והוספו נקודות לאחר בדיקה מעמיקה של התשובות.

ג. (3 נקודות) בתנועות T_1, T_2, \dots, T_n יש רק פעולות קריאה.

תמיד שקולים, באופן ריק, כי אין קונפליקטים בין פעולות קריאה.

ד. (3 נקודות) בתנועות T_1, T_2, \dots, T_n יש רק פעולות כתיבה.

אולי כן (אם פעולות הכתיבה הם למקומות שונים ולכן אין בכלל קונפליקטים בתזמונים הללו), ואולי לא, הטרנסאקציות כותבות לאותם מקומות.

ב. (3 נקודות) האם פרוטוקול שמקיים את התנאים הבאים (1-3) מבטיח תזמונים בני-סידור קונפליקט? נמקו בקצרה.

1. הטרנסאקציה מבקשת מנעולי קריאה (משותפים) עבור כל האלמנטים אליהם תיגש (לקריאה או

לכתיבה) במהלך הריצה שלה, ורק לאחר שקיבלה את כל המנעולים היא מתחילה לפעול.

2. בפעם הראשונה בה הטרנסאקציה צריכה לכתוב לאלמנט כלשהו, היא משדרגת את המנעול למנעול כתיבה (אקסקלוסיבי).

3. מנעולים משוחררים לפי פרוטוקול two-phase locking.

הפרוטוקול מבטיח תזמונים serializable, משום שהוא גרסה חזקה יותר של two-phase locking.

ג. (4 נקודות) האם פרוטוקול strict two-phase locking מונע dirty read? נמקו בקצרה או תנו דוגמה נגדית.

strict two-phase locking מונע dirty reads כי המנעולים משוחררים רק לאחר שברור אם הטרנסאקציה התחייבה ופעולות הכתיבה שלה בוצעו, או לא.