



הטכניון - מכון טכנולוגי לישראל
הפקולטה למדעי המחשב

פרופ' ח' אלדר פיושר
נדב שרגאי

אביב תשס"ח
30, מאי 2008

מערכות מסדי נתונים – 236363

מועד ב'
הזמן: 3 שעות
במבחן זה 12 עמודים

עדכונים על הלוח במהלך הבחינה והערות על הבדיקה כתובים באדום

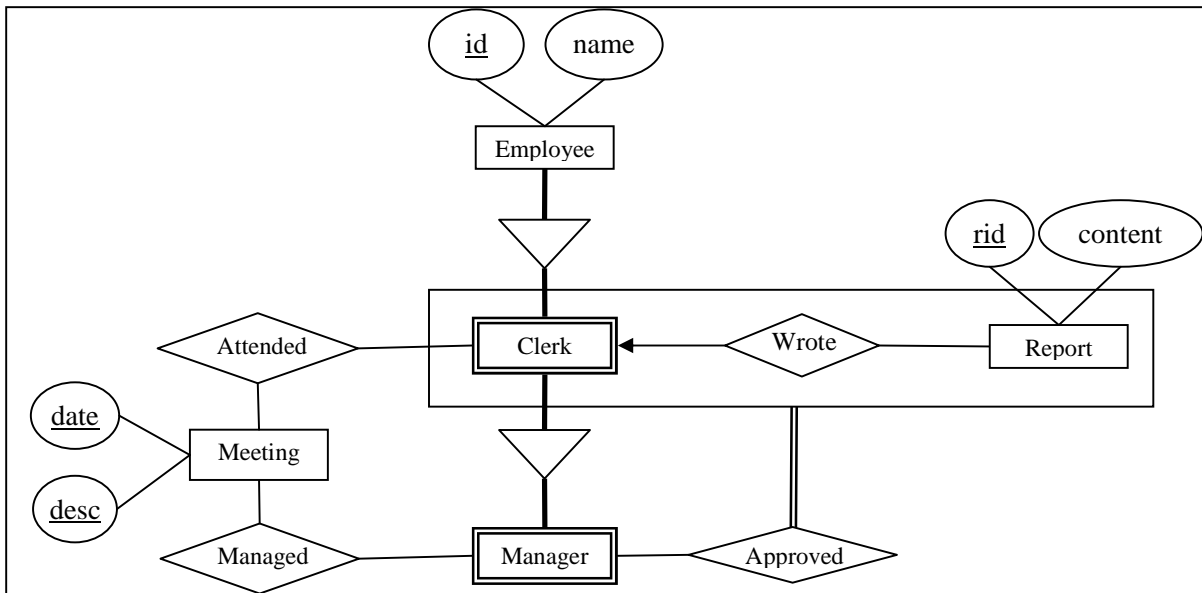
שאלה	נקודות
שאלה 1 – ERD	22
שאלה 2 – שפות שאילתא	31
שאלה 3 – תלויות	29
שאלה 4 – XML	18
סה"כ	100

הנחיות:

- יש לענות על כל השאלות **בטופס הבחינה** מחברות הטיוטה לא תיאספנה.
 - חומר עזר מותר: רק דברים שעשויים מנייר.
 - אין להחזיק מכשיר אלקטרוני כלשהו לרבות מחשב כיס.**
 - קראו היטב את ההוראות שבתחילת כל שאלה ואת ההסברים לסכמות.
 - מומלץ שתכננו היטב את זמנכם, **לא תינתנה הארכות.**
 - ערעורים יש להגיש תוך שבועיים ממועד פרסום התוצאות.
 - לא יתקבלו ערעורים בנוסח "בדיקה מחמירה מדי".
- בהצלחה

שאלה 1 – ERD (22 נק')

נתונה סכמת ה-ERD הבאה, המייצגת מערכת בירוקרטית.



הסבר לסכמה:

ישויות:

Report – ישות המתארת דו"חות במערכת. דו"חות מאופיינים ע"י מספר סידורי (rid) וע"י התוכן שלהם (content)

Employee – ישות המתארת עובדים במערכת. עובדים מאופיינים ע"י מספר זהות (id) וע"י שמם (name). עובדים יכולים להיות פקידים (**Clerk**) ופקידים יכולים להיות מנהלים (**Manager**).

Meeting – ישות המתארת פגישות במערכת. פגישות מאופיינות ע"י תאריך (date) וע"י תאור (desc).

יחסים:

Wrote – יחס המתאר כתיבת דו"ח ע"י פקיד מסוים.

Approved – יחס המתאר אישור דו"ח ע"י מנהל (במידת הצורך השתמשו ב-cid עבור id של Clerk וב-mid עבור id של Manager).

Attended – יחס המתאר נוכחות של פקיד בפגישה.

Managed – יחס המתאר ניהול פגישה ע"י מנהל.

הערה: למעשה את Approved ראוי היה לממש רק עם ה-id של Manager (בגלל החץ מ-Wrote), אבל עקב ההטעיה כאן ובשאלה 2 התקבלו גם תשובות המניחות מימוש עם שני ה-id.

א. (6 נק') מלאו את הטבלה הבאה שמתייחסת לשדות ולמפתחות (הקבילים) של יחסים בסכמת ה-ERD.

Attended	
שדות	date, desc, id
מפתחות	date, desc, id
Manager	
שדות	id, name
מפתחות	id
Wrote	
שדות	id, rid
מפתחות	rid

ב. (8 נק') האם כל דו"ח בסכמת ה-ERD חייב להיות מאושר? נמקו אם כן, ואחרת תארו במילים איך אפשר לשנות את ה-ERD כך שזה יהיה המצב.

לא. ייתכן דו"ח שלא ישתתף באף קשר מסוג Wrote ואז הוא לא יהיה מאושר.

פתרון 1

צריך להוסיף קו כפול מ-Report ל-Wrote כך שכל דו"ח בהכרח נכתב ע"י Clerk ולכן מופיע בהקבצה של כתיבת דו"ח.

מאחר ויש קו כפול בין הקבצה של כתיבת דו"ח לבין Approved, כל הקבצה של כתיבת דו"ח חייבת להיות מאושרת ולכן גם הדו"ח יהיה חייב להיות מאושר.

פתרון 2

יש לשנות את היחס Approved כך שיתחבר ל-Report ולא להקבצה של Wrote. כך כל דו"ח חייב להיות מאושר.

הערה: השאלה איזה פיתרון עדיף תלויה בשאלה האם אנו מוכנים להוסיף תנאי חדש שדו"ח יהיה כתוב (פתרון 1) או לוותר על הקשר בין אישור לבין כתיבה (פתרון 2).

ג. (8 נק') תארו במילים פשוטות אילו שינויים יש לעשות בסכמה המקורית, כך שה-ERD החדש יענה על הדרישות הבאות:

- i. נוסף לפקידים יש במערכת גם מזכירות.
- ii. כל העובדים הם או פקידים או מזכירות.
- iii. גם מזכירות יכולות לכתוב דו"חות.
- iv. רק פקידים יכולים להיות מנהלים.
- v. רק פקידים נוכחים בפגישות.

יש לוודא שהישות Report תישאר מקושרת רק לטיפוס קשר יחיד.

יש להוסיף ישות Secretary.
יש להחליף את קשר הזיהוי מ-Employee ל-Clerk בקשר הכללה מ-Employee ל-Clerk ול-Secretary.
לבסוף יש לקשר את Wrote ל-Employee במקום ל-Clerk ולשנות את ההקבצה להכיל את Employee במקום את Clerk.

שאלה 2 – שפות שאילתא (31 נק')

הערה: בסעיפים הבאים הניחו תרגום סטנדרטי של סכמת ה-ERD המקורית משאלה 1 לסכמות רלציוניות: Report, Employee, Clerk, Manager, Meeting, Wrote, Approved, Attended, Managed.
בשאלות העוסקות ב-DRC הניחו סדר אלפביתי על התכונות של הרלציות.

- א. (12 נק')
i. (8 נק') צרו ב-SQL מבט בשם Honest שמחזיר את ה-id וה-name של כל המנהלים הישרים. מנהל יקרא ישר אם כל הדו"חות שלו אושרו ע"י מנהלים אחרים (אפשריים דו"חות שאושרו במשותף על ידו ומנהלים אחרים).
דוחות של המנהל הינם דוחות שהמנהל כתב

```
CREATE VIEW Honest as (  
  SELECT *  
  FROM Manager m  
  WHERE id NOT IN (  
    SELECT id  
    FROM Wrote  
    WHERE rid NOT IN (  
      SELECT rid  
      FROM Approved  
      WHERE m.id <> mid  
    )  
  )  
)
```

ii. (4 נק') כתבו שאילתה ב-SQL אשר משתמשת במבט Honest מסעיף i על מנת להחזיר את היחס בין מספר המנהלים הישרים לבין מספר המנהלים הלא ישרים. הניחו שקיימים מנהלים לא ישרים. (ניתן להניח שהמבט Honest קיים גם אם לא פתרתם את סעיף i).

אין להשתמש במבטי עזר פרט ל-Honest

```
SELECT 1.0*(
  SELECT COUNT (*) FROM Honest
)/(
  SELECT COUNT (*)
  FROM Manager
  WHERE id NOT IN (SELECT id FROM Honest)
)
```

ב. (6 נק') כתבו שאילתת DRC אשר מחזירה מנהלים (מספיק להחזיר את הזיהוי id שלהם) שהשתתפו באותו יום בלפחות שתי פגישות כשבאחת מהן הם היו מנהלי הפגישה ובשנייה השתתפו כנוכחים.

```
{ <id> |  $\exists$  name (Manager(id, name)  $\wedge$   $\exists$  date, desc1, desc2 (
  Menaged(date, desc1, id)  $\wedge$  Attended(date, desc2, id)  $\wedge$  desc1  $\neq$  desc2)) }
```

ג. (6 נק') נתונה תוכנית Datalog המשתמשת בפרדיקטים מפורשים (EDB) ע"פ הסכמות הרלציוניות הנובעות מה-ERD.

$A(y,z) \leftarrow \text{Attended}(x_1, x_2, y), \text{Managed}(x_1, x_2, z)$

$\text{Query}(z) \leftarrow A(y,z), \text{Approved}(y,z,x)$

כתבו במילים פשוטות מה מחזירה השאילתה $\text{Query}(z)$?

השאילתה תחזיר מספרי זהות של מנהלים אשר קיים עבורם פקיד שהם גם אישרו לו דו"ח וגם ניהלו ישיבה שבה הוא השתתף.

ד. (7 נק') לסעיף זה אין קשר לסעיפים הקודמים.
כתבו שאילתת RA אשר בהינתן רלציה r מעל הסכמה $R(A,B,C)$ תחזיר האם היא אינה מקיימת את התלות הרב ערכית $A \twoheadrightarrow C$. יש להחזיר רלציה חסרת תכונות עם שורה ריקה אם התלות לא מתקיימת, ואחרת יש להחזיר רלציה חסרת תכונות וריקה (ללא שורות).

$$\pi_{\lambda}(\sigma_{a_1=a_2}(r \times r) \setminus \pi_{r_1 r_2}(\sigma_{a_1=a_2 \wedge a_2=a_3 \wedge c_1=c_3 \wedge b_3=b_2}(r_1 \times r_2 \times r_3)))$$

שאלה 3 – Design (29 נק')

א. (12 נק') עבור השאלות הבאות אם התשובה שלילית הסבירו בקצרה ואם היא חיובית תנו דוגמא פשוטה ככל שניתן.
השאלות דנות בסכמה R עם קבוצת תלויות פונקציונליות F.

i. (4 נק') האם ייתכן של-R יהיו יותר מפתחות קבילים מעל-מפתחות?

לא. כל מפתח קביל הוא גם על-מפתח.

ii. (4 נק') האם ייתכן של-R יהיו יותר מפתחות קבילים מאשר תכונות?

כן. למשל הסכמה $R(A,B,C,D)$ עם קבוצת התלויות הפונקציונליות שעבורה כל זוג תכונות קובע את זוג התכונות המשלים (ראו פירוט למטה). התוצאה היא שכל זוג תכונות הוא מפתח קביל.
 $F = \{AB \rightarrow CD, AC \rightarrow BD, AD \rightarrow BC, BC \rightarrow AD, BD \rightarrow AC, CD \rightarrow AB\}$

iii. (4 נק') האם ייתכן של-R יהיו מספר מפתחות קבילים בגדלים שונים זה מזה?

כן. בסכמה $R(A,B,C)$ עם קבוצת התלויות הפונקציונליות $F = \{A \rightarrow BC, BC \rightarrow A\}$ המפתחות הקבילים הם: A, BC

ב. (9 נק') נתונה הסכמה $R(A,B,C,D)$ וקבוצת התלויות הפונקציונליות $F = \{A \rightarrow C, BC \rightarrow D, D \rightarrow B\}$. ענו על השאלות הבאות:

i. (3 נק') בהינתן שברלציה r מעל הסכמה R יש n ערכי A , כמה ערכי C לכל היותר יש ב- r ? נמקו.

C תלוי פונקציונלית ב- A . משמע שעבור כל ערך של A יש לכל היותר ערך אחד של C .
מכך שעבור n ערכים של A יהיו ל- C לכל היותר n ערכים.

ii. (3 נק') בהינתן שברלציה r מעל הסכמה R יש m ערכי B ו- p ערכי C כמה ערכי D לכל היותר יש ב- r ? נמקו.

D תלוי פונקציונלית ב- BC . משמע שעבור כל ערך של BC יש לכל היותר ערך אחד של D .
יש m ערכי B ו- p ערכי C ולכן יש לכל היותר mp ערכי BC .
מכאן שעבור m ערכי B ו- p ערכי C נקבל לכל היותר mp ערכי D .

iii. (3 נק') בהינתן שברלציה r מעל הסכמה R יש q ערכי B , כמה ערכי D לפחות יש ב- R ? נמקו.

B תלוי פונקציונלית ב- D ולכן כל ערך של D קובע ערך יחיד של B .
לכן עבור q ערכי B נדרשים q ערכי D לפחות (כי לא ייתכן שערך D אחד ייקבע יותר מערך B אחד).

ג. (8 נק') נתונה הסכמה $R(A,B,C,D,E)$ וקבוצת התלויות הפונקציונליות
 $F=\{BD \rightarrow AE, A \rightarrow DC, C \rightarrow B, B \rightarrow C\}$

מצאו פירוק של R ב-3NF המשמר מידע ותלויות ומכיל לכל היותר 3 תתי-סכמות.

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

F היא מינימאלית ולכן זהו בדיוק לפי האלגוריתם מהתרגול (ז"א שהפירוק מקיים את הדרוש).

שאלה 4 – XML (18 נק')

במהלך שאלה זו התייחסו לקובץ k.xml המתאר פרטוקולים של פגישות. הקובץ הוא בעל צומת מסמך all כבן יחידי של השורש, ומציית ל-DTD הבא:

```
<!ELEMENT all (meet)*>
<!ELEMENT meet (title,date,part+,phrase+)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT date (#PCDATA)>
<!ELEMENT part (#PCDATA)>
<!ELEMENT phrase (part,text)>
<!ELEMENT text (#PCDATA)>
```

הסבר: לכל פגישה meet יש את צמתי הבנים הבאים:

title – כותרת הפגישה.

date – תאריך הפגישה.

part אחד או יותר – כל צומת כזה מכיל שם של משתתף.

phrase אחד או יותר – כל צומת כזה מתאר טיעון שהועלה בפגישה (הצמתיים מסודרים לפי סדר הדיבור שהיה בפגישה). בניו הם שם הדובר part ותוכן הטיעון text.

א. (8 נק') כתבו שאילתת XPath 1.0 אשר תמצא משתתפים אילמים בפגישות. משתתף אילם הוא אחד ששמו הופיע כמשתתף אך שלא העלה אף טיעון בפגישה. יש להחזיר את הצמתיים המתאימים למשתתפים האילמים בכל פגישה ופגישה.

```
//meet/part[not(. = ../phrase/part)]
```

הערה: טעות נפוצה היתה להתחיל את הביטוי ב-part // (ללא חיפוש meet), אבל אז יוחזרו בטעות גם צמתי part של הטיעונים (שבוודאי אינם מציינים משתתפים אילמים) ולא רק של הפגישה עצמה.

ב. (10 נק') לאדם אשר השתתף בפגישה אחת לפחות נקרא **עובד**, ולתאריך שהתקיימה בו פגישה כל שהיא נקרא **יום עבודה**. כתבו שאילתת XQuery 1.0 אשר לכל עובד מוציאה את **מספר ימי העבודה שבהם הוא לא השתתף באף פגישה** (גם לא כמשתתף אילם). אסור שתהיינה כפילויות בשמות. על השאילתה להוציא מסמך XML בנוי היטב, ועליכם לתאר את הסכימה של קובץ הפלט (אפשר לתאר אותה במילים או בצורת DTD).

```
xquery version "1.0";
let $doc:=doc("p.xml")
return
<list>
{
  for $p in distinct-values($doc//meet/part)
  return
    <part name="{ $p}" num="{ count(
      for $date in distinct-values($doc//date)
      where not ($doc//meet[part=$p and date=$date])
      return $date
    )}"/>
}
</list>
```

הסכימה של קובץ הפלט בצורת DTD:
list הוא צומת המסמך כבן יחיד של השורש

```
<!ELEMENT list (part*)>
<!ELEMENT part EMPTY>
<!ATTLIST part name CDATA #REQUIRED
              num CDATA #REQUIRED>
```

הערה: אפשר להחליף את הביטוי בתוך פונקציית ה-count בביטוי פשוט יותר המשתמש בהפרש של קבוצות:
distinct-values(\$doc//date) except (\$doc//meet[part=\$p]/date)

אפשרות שניה היא לחשב מראש את מספר התאריכים השונים (עם let בהתחלה) ואחר כך לעבוד עם חיסור נומירי של ספירת התאריכים המתאימה.