



הטכניון - מכון טכנולוגי לישראל
הפקולטה למדעי המחשב

אביב תשס"ח
30, אוקטובר 2008

ד"ר אמיר שפילקה
נדב שרגאי

מערכות מסדי נתונים – 236363

מועד ב'
הזמן: 3 שעות
במבחן זה 13 עמודים

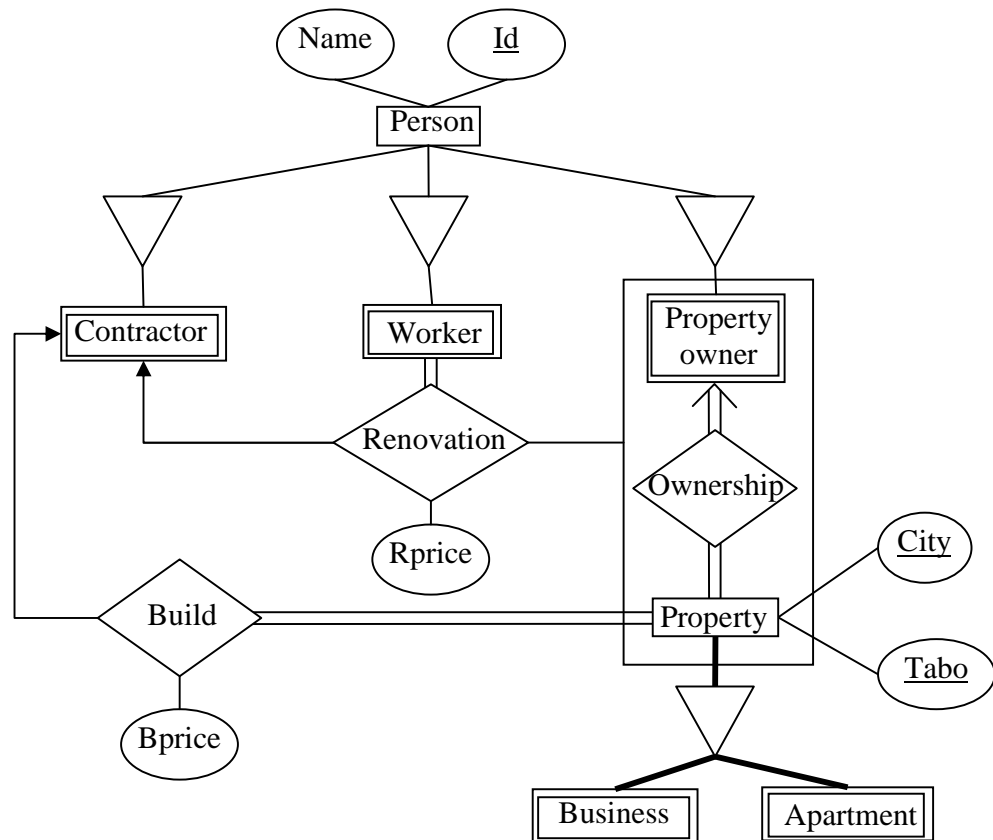
שאלה	נקודות
שאלה 1 – ERD	18
שאלה 2 – שפות שאלתא	36
שאלה 3 – פירוקים וצורות נורמליות	23
שאלה 4 – XML	23
סה"כ	100

הנחיות:

- יש לענות על כל השאלות **בטופס הבחינה** מחברות הטייטה לא תיאספנה.
 - חומר עזר מותר: דף נוסחאות A3 יחיד מודפס משני הצדדים.
 - 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 (18 נק')

נתונה סכמת ה-ERD הבאה המתארת נכסים, הבעלים שלהם והקבלנים שבנו או שיפצו את הנכס.



הסבר לסכמה:

טיפוסי ישות:

- Person** - ישות המתארת אנשים. אנשים מאופיינים ע"י מזהה ייחודי (ID) ושם (Name). אנשים יכולים להיות קבלנים (Contractor) ו/או פועלים (Worker) ו/או בעלי נכסים (Property owner).
- Property** - ישות המתארת נכס. נכס מאופיין ע"י עיר (City) ונכס טאבו (Tabo). נכס יכול להיות עסק (Business) או דירה (Apartment).

טיפוסי קשר:

- Ownership** - יחס המתאר בעלות של בעל נכסים על נכס מסוים.
 - Build** - יחס המתאר בנית נכס ע"י קבלן במחיר מסוים (BPrice).
 - Renovation** - יחס המתאר שיפוץ שנעשה לנכס ע"י קבלן ופועל במחיר נתון (RPrice).
- במידת הצורך על מנת להבדיל בין השדות של קבוצות ישויות שונות השתמשו לפני name ו-id בקידומות Po, W, C, Pe עבור איש, קבלן, פועל ובעל נכסים בהתאמה (לדוגמא Peid עבור id של איש).

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

Renovation	
שדות	Cid, Wid, City, Tabo, Rprice
מפתחות	Wid, City, Tabo

Build	
שדות	Cid, City, Tabo, Bprice
מפתחות	City, Tabo

Apartment	
שדות	City, Tabo
מפתחות	City, Tabo

ב. (6 נק') ענו על השאלות הבאות המתייחסות ל-ERD הנתון. נמקו את תשובותיכם.
 i. (3 נק') האם ייתכן יותר מקבלן אחד שמשפץ נכס נתון?

כן.
 החץ בין שיפוץ לבין קבלן אומר שבהינתן נכס ופועל הקבלן נקבע באופן יחיד.
 לכן עבור נכס נתון יתכנו כמה פועלים כשכל פועל עובד עבור קבלן אחר שמשפץ את הנכס.

ii. (3 נק') האם ייתכן שבעל ואישה יהיו שניהם בעלי אותו נכס?

לא.
 החץ בין נכס לבין בעל אומר שעבור כל נכס יש בעלים יחיד.
 משום כך לא ייתכן ששני אנשים יהיו בעלים של אותו נכס.
 בפרט גם בעל ואישה אינם יכולים להיות בעלים של אותו נכס.

ג. (6 נק') על פי ה-ERD פועל יכול לעבוד על נכס מסוים רק עם קבלן יחיד.
הסבירו כיצד לשנות את ה-ERD כך שאם נרצה לבצע שיפוץ נוסף לאותו נכס עם אותו
פועל נוכל להשתמש בקבלן אחר.

נוסיף שדה תאריך עם קו תחתון לטיפוס קשר של שיפוץ.
כך נוכל להבדיל בין שיפוצים שונים ע"י תאריך השיפוץ ובכל שיפוץ נבחר בקבלן שאנו רוצים
שיעבוד עם הפועל הנתון על הנכס הנתון.

שאלה 2 – שפות שאילתא (36 נק')

הערה: בסעיפים הבאים הניחו תרגום סטנדרטי של סכמת ה-ERD המקורית משאלה 1 לסכמות רלציוניות: **Pe** = Person, **C** = Contractor, **W** = Worker, **Po** = Property owner, **Pr** = Property, **A** = Apartment, **Bs** = Business, **R** = Renovation, **O** = Ownership, **Bd** = Build. בשאלות העוסקות ב-DRC הניחו סדר אלפביתי על התכונות של הרלציות.

א. (7 נק') נאמר שקבלן הוא קבלן דירות שעובד בעיר מסוימת אם הוא שיפץ בעיר זאת לפחות דירה אחת (ללא תלות במספר העסקים ששיפץ).
המחיר של קבלן דירות בעיר מסוימת הוא המחיר הממוצע של כל הדירות ששיפץ באותה עיר. קבלן דירות הוא קבלן הדירות הכי זול בעיר מסוימת אם המחיר שלו באותה עיר הוא המחיר הכי זול מבין כל קבלני הדירות באותה העיר.

כתבו מבט CheapContractor(City, Cid, price) אשר מחזיר לכל עיר את קבלן הדירות הכי זול בה ואת המחיר שלו (אם יש יותר מקבלן זול אחד יש להחזיר את כולם).
ניתן להשתמש במבטי עזר

```
CREATE VIEW prices AS
SELECT R.City, R.Cid, AVG(R.Rprice) as price
FROM R
WHERE (R.City, R.Tabo) IN (
    SELECT A.City, A.Tabo
    FROM A
)
GROUP BY City, Cid;

CREATE VIEW CheapContractor AS
SELECT *
FROM prices p
WHERE (p.City, p.price) IN (
    SELECT p2.City, MIN(p2.price)
    FROM prices p2
    GROUP BY City
);
```

ב. (7 נק') נגדיר "פראיירים" כאנשים ששיפצו את כל הדירות שלהם בעיר מסוימת, ועבור כל אחד מהשיפוצים שילמו לפחות 10% יותר מהמחיר של קבלן הדירות הזול ביותר באותה העיר. החזירו לכל עיר את מספר ה"פראיירים" שבה (שימו לב שאותו "פראייר" יכול להיות ולהיספר בכמה ערים שונות).
הניחו כי CheapContractor(City, Cid, price) הוא המבט שחושב בסעיף הקודם.

```
SELECT s.City, COUNT(s.Id)
FROM (
    SELECT O.City, O.Id
    FROM O, R, A
    WHERE O.City = R.City AND R.City = A.City AND O.Tabo = R.Tabo AND
        R.Tabo = A.tabo
    GROUP BY O.City, O.Id
    HAVING MIN(Rprice) >= 1.1*(
        SELECT price
        FROM CheapContractor c
        WHERE c.City = O.City
    )
) s
GROUP BY s.City;
```

ג. (7 נק') כתבו שאילתא ב-DRC אשר מחזירה את כל הקבלנים (ID בלבד) שיש בבעלותם לפחות עסק אחד אותו הם בנו בעצמם אבל נתנו לקבלן אחר לשפץ.

$$\{ \langle x \rangle \mid \exists n C(x, n) \wedge \exists c, t (Bs(c, t) \wedge O(c, x, t) \wedge \exists p_1 Bd(p_1, c, x, t) \wedge \exists y, w_{id}, p_2 ((y \neq x) \wedge R(c, y, p_2, t, w_{id}))) \}$$

ד. (7 נק') כתבו שאילתא ב-RA אשר מחזירה את כל בעלי הנכסים (ID בלבד) ששיפצו את כל הדירות שלהם אך לא שיפצו אף אחד מהעסקים שלהם (כולל המקרה הריק של בעלי נכסים ללא עסקים ו/או ללא דירות).

$$Q1 = \pi_{id}\{O \times [A \setminus (A \times R)]\}$$

Q1: כל בעלי הדירות שלא שיפצו אחת הדירות שבבעלותם

$$Q2 = \pi_{id}[O \times (B \times R)]$$

Q2: כל בעלי העסקים ששיפצו איזשהו עסק

$$Ans = \pi_{id}(Po) \setminus (Q1 \cup Q2)$$

ה. (8 נק') נתון ה-EDB הבא: $p(X)$ המכיל אנשים, $vip(X)$ המכיל אנשים חשובים (כמובן $vip \subseteq p$) ו- $r(X,Y)$ האומר ש- X שיפץ נכס עבור Y . נאמר כי אנשים A ו- B מקושרים אם A בנה או שיפץ נכס בבעלות B או להפך. היינו, אם מתקיים $r(A,B)$ או $r(B,A)$. בנוסף נאמר כי אנשים A ו- B מקושרים, אם יש אדם C כך ש- A מקושר ל- C ו- B מקושר ל- C . בנוסף, כל אדם מקושר לעצמו. כתבו תוכנית Datalog המחזירה את זוגות האנשים החשובים הלא מקושרים.

$$link(X,Y) \leftarrow r(X,Y)$$

$$link(X,Y) \leftarrow r(Y,X)$$

$$link(X,Y) \leftarrow link(X,Z), link(Y,Z)$$

$$link(X,X) \leftarrow p(x)$$

$$nolink(X,Y) \leftarrow \neg link(X,Y), vip(X), vip(Y)$$

שאלה 3 – פירוקים וצורות נורמליות (23 נק')

א. (6 נק') יהי $R[U]$ יחס עם קבוצת תלויות F . נאמר כי שתי תתי קבוצות $X, Y \subseteq U$ הן דומות אם לכל תלות $X \rightarrow Z$ ב- F^+ מתקיים שגם $Y \rightarrow Z$ ב- F^+ ולהפך. בהינתן X, Y האם יש אלגוריתם יעיל הבודק האם X ו- Y דומות? נמקו.

כן. נשים לב כי אם X ו- Y דומות אז מהתלות $Y \rightarrow Y$ נקבל גם $X \rightarrow Y$ ובאותו אופן נקבל גם $Y \rightarrow X$. כמו כן ברור כי אם מתקיים $X \rightarrow Y$ ו- $Y \rightarrow X$ אז בודאי X ו- Y דומות (מטרנזיטיביות). לכן כל שצריך לבדוק הוא האם מתקיים $X \rightarrow Y$ ו- $Y \rightarrow X$ וזה ניתן לעשות בקלות, למשל ע"י חישוב X_F^+ ו- Y_F^+ כפי שנלמד בכיתה ובדיקה שאכן $X \subseteq Y_F^+$ ו- $Y \subseteq X_F^+$.

ב. (17 נק) נתונה הרלציה $R(A, B, C, D, E, H)$ וקבוצת התלויות הפונקציונאליות $F = \{BH \rightarrow A, E \rightarrow AB, HD \rightarrow AB, BC \rightarrow H, D \rightarrow AE, C \rightarrow D\}$.

י. (5 נק) מצאו כיסוי ל- F .

$\{E \rightarrow A, E \rightarrow B, BH \rightarrow A, C \rightarrow H, D \rightarrow E, C \rightarrow D\}$

ii. (4 נק) האם הפירוק $R_1(A,B,D,H)$, $R_2(C,D,E,H)$, $R_3(B,E)$ משמר תלויות? נמקו.

לא.
ניתן לראות שהתלות $E \rightarrow AB$ אינה נשמרת בפירוק.

iii. (4 נק) האם הפירוק הנ"ל משמר מידע? נמקו.

כן.
נפרק את $R(A,B,C,D,E,H)$ ל- $R(A,B,C,D,E,H)$ ול- $R_3(E,B)$. כמובן ש- R_3 נובע מהחיתוך EB ולכן זה פירוק משמר מידע.
נמשיך ונפרק את $R(A,B,C,D,E,H)$ ל- $R_1(A,B,D,H)$ ול- $R_2(C,D,E,H)$. החיתוך הוא HD וע"פ התלות $HD \rightarrow AB$ רואים ש- R_2 נובעת מהחיתוך ולכן זה פירוק משמר מידע.
מכך שהגענו לפירוק R_1, R_2, R_3 בסדרת פירוקים משמרי מידע ולכן כל הפירוק משמר מידע.

iv. (4 נק) מצאו פירוק משמר מידע ומשמר תלויות ל-R.

הפירוק הוא $R_1(A,B,D,H)$, $R_2(C,D,E,H)$, $R_3(A,B,E)$.
בסעיף ii רואים ש- $E \rightarrow AB$ היא התלות היחידה שאינה נשמרת.
לכן הוספת A ל- R_3 משמרת את התלות הזאת.
בפתרון בסעיף iii ניתן להחליף את A,B,E במקום B,E והפתרון יישאר תקף.
מכאן שהפירוק משמר גם מידע וגם תלויות.

שאלה 4 – XML (23 נק')

נתון קובץ XML המציית ל- DTD הבא:

```
<!ELEMENT RealEstate (person*, property*) >
<!ELEMENT person EMPTY>
<!ATTLIST person
  id ID #REQUIRED
  name CDATA #REQUIRED
>
<!ELEMENT property (project)*>
<!ATTLIST property
  address ID #REQUIRED
  owner IDREF #REQUIRED
>
<!ELEMENT project EMPTY>
<!ATTLIST project
  type (renovation|build) "renovation"
  contractor IDREF #REQUIRED
>
```

צומת המסמך הוא RealEstate והוא מכיל אנשים (person) ונכסים (property).
אנשים מאופיינים ע"י מספר זיהוי (id) ושם (Name).
נכסים מאופיינים ע"י כתובת (address) ו-בעלים (owner) המצביע ל-person, ומכילים פרויקטים (project).
פרויקטים מאופיינים ע"י סוג הפרויקט (type) שזה שיפוץ (renovation) או בנייה (build) והקבלן (contractor) המצביע ל-person.

שימו לב שה-DTD דומה ל-ERD שמופיע בשאלה 1 אך אינו זהה לו.

א. (9 נק') כתבו שאילתת XPATH המחזירה את כל הקבלנים כך שאם הם שיפצו אצל מישהו נכס אז כל שיפוץ נוסף שהוא עשה באותו הנכס (מאוחר יותר) התבצע אצלם.
הניחו שהפרויקטים של נכס כתובים במסמך ע"פ סדר הביצוע שלהם (בין השאר: הפרויקט הראשון הוא תמיד בנייה (build) ואין בנכס פרויקט בנייה נוסף)
ניתן להתעלם ממקרים ריקים למיניהם (כלומר נתון לבחירתכם האם לכלול אותם בבחירה או לא)

```
id(//@contractor[../@type="renovation"]
  not (.= ../following-sibling::*/@contractor) and
  not (.= ../preceding-sibling::*[type="renovation"]/@contractor)
])
```

ב. (8 נק') החזירו את כתובות כל הנכסים כך שהבעלים שלהם גם בנו או שיפצו אותם לפחות פעם אחת (כקבלנים)

```
//@address[../@owner = ../project/@contractor]
```

ג. (6 נק') לפניכם קטע ממסמך XML שאינו תקף ביחס ל-DTD הנ"ל.

```
<RealEstate>
  <person id="p1" name="nadav" secondName="Joshua"></person>
  < person id="p2" name="tamar" middleName="Miriami"></person>
  <property owner="p1" address="motzkin">
    < person id="p3" name="shmuel"></person>
    <project type="build" contractor="p3"></project>
    <project type="renovation" contractor="p3"></project>
    <project contractor="p2"></project>
  </property>
  <property owner="p1" address="12th_street_9_Philadelphia">
    <project type="build" contractor="p1"></project>
    <project type="expansion" contractor="p2"></project>
  </property>
  <property owner="p2" address="Broadway_5_New_York"></property>
</RealEstate>
```

שנו את מסמך ה-DTD הנ"ל על מנת שהמסמך יהיה תקף.
הסבירו כל אחד מהשינויים.
לנוחותכם, מצורף להלן ה-DTD המקורי. מותר להשאיר או לתקן ו/או למחוק שורות ישנות, וכן מותר להוסיף שורות חדשות.

<!ELEMENT RealEstate (person*, property*) >

<!ELEMENT person EMPTY>

<!ATTLIST person

id ID #REQUIRED

name CDATA #REQUIRED

middleName CDATA #IMPLIED

secondName CDATA #IMPLIED

השינוי הזה הוא על מנת לאפשר את שתי שורות person הראשונות.

>

<!ELEMENT property (person|project)*>

השינוי הזה הוא על מנת לאפשר את הופעת אלמנט person בשורה החמישית בתוך אלמנט property.

<!ATTLIST property

address CDATA #REQUIRED

השינוי הזה הוא עבור ה-address של ה-property השני שאינו מתחיל באות כפי שצריך להיות.

owner IDREF #REQUIRED

>

<!ELEMENT project EMPTY>

<!ATTLIST project

type (renovation|build|expansion) "renovation"

השינוי הזה הוא עבור ה-project השני ב-property השני. שם הסוג של ה-project הוגדר כ-expansion.

contractor IDREF #REQUIRED

>