



פרופסור עודד שמואלי
מר סעאב מנסור
גברת לינה זריבץ'

חורף תשס"ו
19, פברואר 2006

מערכות מסדי נתונים - 236363 בחן 2 - מועד א'

הזמן: 105 דקות (1:45 שעות)
במבחן זה 8 עמודים

שם פרטי: _____

שם משפחה: _____

מס' סטודנט: _____

פקולטה: _____

שאלה	נקודות	מתוך
שאלה 1		36
שאלה 2		32
שאלה 3		16
שאלה 4		16
סה"כ		100

הערות:

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

בהצלחה

שאלה 1 – Design (36 נק')

א. (18 נק') נתונה סכמה R וקבוצת תלויות פונקציונליות F. נתונים פרקים ρ_1, ρ_2 של R. האם ייתכן המצב המתואר בסעיפים הבאים. אם לא הוכח זאת, אם כן תן דוגמא קצרה ובה ל-R יש לכל היותר ארבעה אטריביוטים.

i. (6 נק') ρ_1 משמר מידע ביחס ל-R, ρ_2 פרוק כלשהו ביחס ל-R, F, $\rho = \rho_1 \cup \rho_2$ אינו משמר מידע ביחס ל-R, F.

מצב זה לא ייתכן. אם נסתכל על צעדי האלגוריתם לבדיקת שימור מידע, אז כיוון ש- ρ_1

משמר מידע, נגיע לשורה ללא אינדקסים, וההוספה של השורות עבור הסכמות ב- ρ_2

אינה מונעת זאת.

ii. (6 נק') ρ_1 משמר מידע ואינו משמר תלויות ביחס ל-R, F, ρ_2 משמר תלויות ואינו משמר מידע ביחס ל-R, F, $\rho = \rho_1 \cup \rho_2$ משמר מידע ומשמר תלויות ביחס ל-R, F.

מצב זה כן ייתכן. למשל $R=\{A,B,C\}$, $F=\{B \rightarrow A, C \rightarrow A\}$, $\rho_1 = \{\{A,B\}, \{B,C\}\}$,

$\rho_2 = \{\{A,B\}, \{A,C\}\}$.

iii. (6 נק') ρ_1 אינו משמר מידע ואינו משמר תלויות ביחס ל-R, F, ρ_2 אינו משמר מידע ואינו משמר תלויות ביחס ל-R, F, $\rho = \rho_1 \cup \rho_2$ משמר מידע ומשמר תלויות ביחס ל-R, F.

מצב זה כן ייתכן. למשל, $R=\{A,B,C,D\}$, $F=\{A \rightarrow B, C \rightarrow D, B \rightarrow C, A \rightarrow D\}$,

$\rho_1 = \{\{A,B\}, \{C,D\}\}$, $\rho_2 = \{\{B,C\}, \{A,D\}\}$.

ב. (18 נק') נתונה הסכמה R,F כאשר $R=\{A,B,C,D,E\}$, $F=\{CE \rightarrow D, D \rightarrow E, ACD \rightarrow B, C \rightarrow A, BC \rightarrow D\}$.

i. (1 נק') האם F מינימאלית? אם כן נמק. אם לא, כתוב כיסוי מינימאלי של F.

F איננה מינימאלית. להלן כיסוי מינימאלי F' שלה:

$F' = \{CE \rightarrow D, D \rightarrow E, CD \rightarrow B, C \rightarrow A, BC \rightarrow D\}$

ii. (3 נק') מהם כל המפתחות הקבילים של R,F?

$\{C,D\}, \{C,B\}, \{C,E\}$

iii. (6 נק') מצא פירוק משמר מידע ומשמר תלויות ל-3NF של R,F ובו בדיוק **שלוש** סכמות.

$\{\{C,A\}, \{C,D,B\}, \{CED\}\}$

iv. (8 נק') מצא פירוק משמר מידע ומשמר תלויות ל-BCNF של R,F ובו בדיוק **ארבע** סכמות.

$\{\{C,E,B\}, \{D,E\}, \{C,A\}, \{B,C,D\}\}$

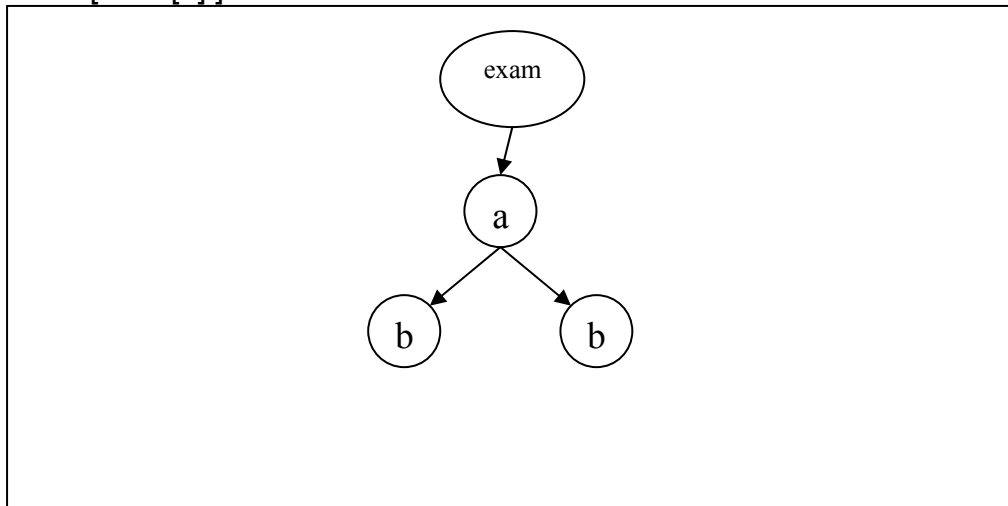
שאלה 2 - XPath (32 נק')

שאלה זאת מתייחסת למסמכי XML המצייתים ל-DTD הבא:

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT exam (a|b|c|d)+>
<!ELEMENT a (a|b|c|d)*>
<!ELEMENT b (a|b|c|d)*>
<!ELEMENT c (a|b|c|d)*>
<!ELEMENT d (a|b|c|d)*>
```

1. (14 נק') קבעו האם עבור כל מסמך XML המציית ל-DTD הנ"ל, שני ביטויי ה-XPath הנתונים בכל סעיף הינם שקולים.
אם כן, הסבר בקצרה מה מחשב כל ביטוי ומדוע החישובים זהים. במידה ולא, יש להראות מסמך XML, בייצוג עצי קצר ככל האפשר, שעליו שני הביטויים מחזירים תוצאות שונות.
א. (7 נק')

1. //b[1]
2. //b[.=../b[1]]



ב. (7 נק')

1. count(//b/*/*)
2. count(//self::node()[parent::node()[name() = "b"]][not(following-sibling::node())])

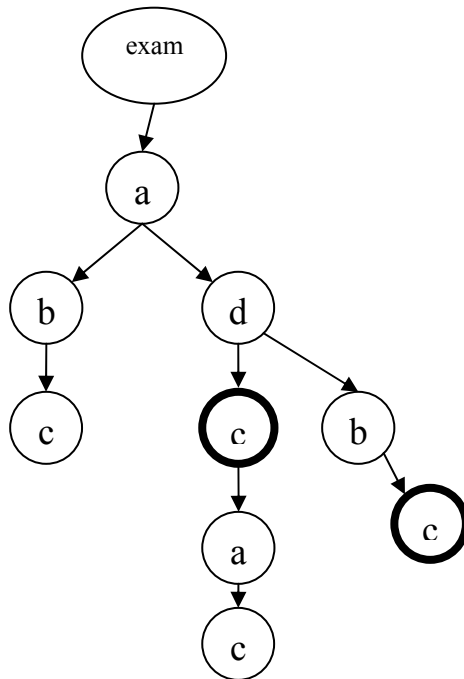
שני הביטויים שקולים.

הביטוי הראשון מחשב את מספר האלמנטים מסוג b שאינם עלים.
הביטוי השני עושה את אותו הדבר ע"י ספירת הבנים האחרונים של b, שזה בעצם מספר ה-b שאינם עלים במסמך.

2. (18 נק') ממשו ב-**XPath** את השאילתות הבאות:

א. (9 נק') לצורך סעיף זה בלבד יש להניח כי בכל מסמך XML יש אלמנט אחד ויחיד מסוג **d**.

כתבו שאילתא ב-XPath המחזירה את כל הצמתים **v** מסוג **c** המקיימים את התנאי הבא: הצומת **v** הינו הצומת הראשון מסוג **c** המופיע במסלול מהצומת **d** לעלה כלשהו. לדוגמא, עבור המסמך המתואר למטה, רק הצמתים המודגשים אמורים להיות מוחזרים.



`//d//c[not(ancestor::c) or count(ancestor::c) = count(ancestor::c[//d])]`

או

`//d//c[not (ancestor::c [ancestor::d])]`

ב. (9 נק') החזירו את האלמנט **exam** אם בכל מסלול בעץ המתחיל מהשורש מופיע צומת **a** אחד לפחות. אחרת, החזירו קבוצה ריקה.

`/*[not(//*[not(*)] [not(ancestor-or-self::a)])]`

שאלה 3 - XQuery (16 נק')

שאלה זאת עוסקת במסמך bib.xml. מסמך זה מציית למסמך ה- DTD הבא (שגם מופיע בשקפי התרגול):

```
<!ELEMENT bib (book)*>
<!ELEMENT book (title, (editor | (author)*), publisher, price)>
<!ATTLIST book
  year CDATA #REQUIRED>
<!ELEMENT title (#PCDATA)>
<!ELEMENT editor (last, first, affiliation)>
<!ELEMENT author (last, first)>
<!ELEMENT publisher (#PCDATA)>
<!ELEMENT price (#PCDATA)>
<!ELEMENT last (#PCDATA)>
<!ELEMENT first (#PCDATA)>
<!ELEMENT affiliation (#PCDATA)>
```

כתבו שאילתא ב- **XQuery** המחזירה עבור כל אלמנט מסוג סופר (author) במסמך, את מספר הספרים שכתב הסופר הנדון (כמחבר בודד או עם סופרים אחרים). השאילתא תוציא לפלט קבוצה של אלמנטים בשם author1 בהם המתארים סופרים ומספר הספרים אשר השתתפו בחיבורם. על האלמנט author1 להכיל את שם הסופר (שם פרטי ושם משפחה) ואטריביוט בשם numBooks המתאר את מספר הספרים שכתב. כמו כן, על התוצאה להיות ממוינת לפי numBooks, בסדר עולה.

לדוגמא, לו bib.xml היה:

```
<bib>
  <book year="1994">
    <title>TCP/IP Illustrated</title>
    <author>
      <last>Stevens</last><first>W.</first>
    </author>
    <author>
      <last>Abiteboul</last><first>Serge</first>
    </author>
    <publisher>Addison-Wesley</publisher>
    <price>65.95</price>
  </book>
  <book year="1992">
    <title>Advanced Programming in the Unix environment</title>
    <author>
      <last>Stevens</last><first>W.</first>
    </author>
    <publisher>Addison-Wesley</publisher>
    <price>65.95</price>
  </book>
</bib>
```

אזי הפלט הצפוי יהיה:

```
<author1 numBooks="1">
  <last>Abiteboul</last>
  <first>Serge</first>
</author1>
<author1 numBooks="2">
  <last>Stevens</last>
  <first>W.</first>
</author1>
<author1 numBooks="2">
  <last>Stevens</last>
  <first>W.</first>
</author1>
```

שימו לב שאם סופר השתתף בכתיבה של N ספרים אז הסופר הנ"ל יופיע N פעמים בפלט.

פתרון:

```
xquery version "1.0";
for $auth in doc("bib.xml")//author
let $num := count($auth/../../book [author = $auth] )
order by $num
return
<author1 numBooks="{ $num }">
  { $auth/* }
</author1>
```

שאלה 4 – DATALOG (16 נק')

נניח ייצוג של מסמך XML כתוכנית DATALOG כפי שנעשה בתרגיל הבית.

תזכורת:

נשתמש בפרדיקטים הבאים כ-EDB:

- Root(ID) – ID is the unique id of the document root.
- ID הנו מזהה ייחודי של שורש המסמך (קיימת עובדה אחת בלבד לפרדיקט Root).
- Tag(ID, TAG) – TAG is the tag of the XML element with unique id ID.
Tag is not defined for the ID of the root.

TAG הנו שם האלמנט של אלמנט ה-XML בעל מזהה ייחודי ID. Tag אינו מוגדר עבור המזהה הייחודי של השורש.

- Edge(ID1, ID2, POS) – The element with ID2 is the POSth child of the element with ID1.

האלמנט בעל מזהה ייחודי ID2 הנו הבן ה-POS במספר (מספר הבן השמאלי ביותר הוא 1) של האלמנט בעל מזהה ייחודי ID1.

- Text(ID, TEXT) – Element ID has a text child element containing the text TEXT.

לאלמנט בעל מזהה ייחודי ID יש בן מסוג text הכולל את המחרוזת TEXT.

לדוגמא (הקיבון לצורך קריאות):

```
<a>
  <b>55</b>
  <b>6</b>
  <d>
    <e>909</e>
  </d>
</a>
```

ייצוג ע"י ה-EDBs הבאים:

Root	Tag	Edge	Text
Root(1).	Tag(2,a).	Edge(1,2,1).	Text(3, 55).
	Tag(3,b).	Edge(2,3,1).	Text(4, 6).
	Tag(4,b).	Edge(2,4,2).	Text(6, 909).
	Tag(5,d).	Edge(2,5,3).	
	Tag(6,e).	Edge(5,6,1).	

כתבו תוכנית DATALOG ובה פרדיקט **lastb(X)** המחשב את ה-IDs של כל האלמנטים

המוחזרים ע"י הביטוי `//b[last()]`.

(תזכורת: last() מחזירה מספר השווה לגודל קבוצת צמתי ההקשר)

`descendant-or-self(X,X) ← Root(X).`

`descendant-or-self(X,Y) ← Edge(X,Z,_), descendant-or-self(Z,Y).`

`lastb(X) ← Root(R), descendant-or-self(R,X), Tag(X,b), ¬bad(X).`

`bad(X) ← Edge(F,X,POS), Edge(F,Y,POS1), Tag(Y,b), POS1>POS.`
