**תרגיל 2 - בסיסי נתונים**

ד"ר רועי קרקובסקי

# כללי

מטרה – בסדרת התרגילים הקרובים, תתנסו בכתיבת תכנית מחשב המממשת את עקרונות הארכיטקטורה והאלגוריתמים של מערכות בסיסי נתונים.

שפה - את התרגילים יש לממש בשפת Python.

**תאריך הגשה:** 27/12/2020

ההגשה בזוגות או ביחידים.

**מטרה** – בהמשך לתרגיל 1 שבו בדקנו את נכונות השאילתה לכללי שפת SQL, בתרגיל זה נרצה לבחון כיצד ניתן לשנות את מבנה השאילתה כך שנוכל להשיג שיפור בביצועים .

בתרגיל נניח את קיומן של שתי סכמות:

**R(A:INTEGER,B:INTEGER,C:INTEGER,D:INTEGER,E:INTEGER)**

**S(D:INTEGER,E:INTEGER,F:INTEGER,H:INTEGER,I:INTEGER)**

## חלק 1 - הפעלת כללי אופטימיזציה

**קלט** – שאילתה תקניתבשפת SQL (לאחר שעברה בדיקת תקינות ע"י תרגיל 1) וכלל אופטימיזציה כלשהו. אין צורך להריץ את הבדיקה של תרגיל, אנחנו נניח שהקלט תקין.

**פלט** – Logical Query Plans המתקבל לאחר הפעלת הכלל הנבחר.

**שלב 1**

ראשית יש להפוך את השאילתה משפת SQL **לביטוי אלגברי** בצורת מחרוזת. למשל:

**PI[R.D,S.E]( SIGMA[S.D>4 AND R.A=10] (** **CARTESIAN( R, S)** **))**

**שלב 2**

אחרי כן עליכם להפעיל את אחד מהכללים **הבאים** בהתאם לבחירת המשתמש:

* 1. **4 (משמאל לימין בלבד)**
  2. **4a**
  3. **6,6a** (ובאופן דומה עם מכפלה קרטזית במקום natural join). **(משמאל לימין בלבד)**
  4. **5a (משמאל לימין בלבד)**
  5. **11b.** **(משמאל לימין בלבד)** את כלל 11b, יש להפעיל רק אם אגף שמאל מתלכד עם אופרטור ה - Natural Join, ואז יש להחליף את אגף שמאל באופרטור ה- Natural Join. מתלכד במובן, שהפרדיקט הוא שוויון על העמודות הזהות בשתי הסכמות. **אנחנו לא תומכים theta join בתרגיל זה.**

יש לקלוט מהמשתמש את הכלל אותו הוא רוצה להפעיל.

לנוחיותכם בסוף מסמך זה הוספתי תזכורת של הכללים. הפלט שלכם אמור להציג את הכלל שהפעלתם ואת הביטוי שהתקבל לאחר הפעלת הכלל. אם כלל לא ניתן להפעלה על הביטוי אז הביטוי נשאר כמו שהוא.

**דוגמא 1**

לדוגמא עבור השאילתה למעלה וכלל **4** נקבל:

**PI[R.D,S.E]( SIGMA[S.D>4]( SIGMA[R.A=10] (** **CARTESIAN( R, S)** **)))**

עבור השאילתה שהתקבלה וכלל **4a** נקבל:

**PI[R.D,S.E]( SIGMA[R.D=10]( SIGMA[S.D>4] (** **CARTESIAN( R, S)** **)))**

עבור השאילתה שהתקבלה וכלל  **6a** נקבל:

**PI[R.D,S.E]( SIGMA[R.D=10](CARTESIAN(R, SIGMA[S.D>4](S))))**

עבור השאילתה שהתקבלה וכלל  **6** נקבל:

**PI[R.D,S.E](CARTESIAN( SIGMA[R.D=10](R), SIGMA[S.D>4](S)))**

**דוגמא 2**

עבור השאילתה:

**PI[R.D,S.E]( SIGMA[S.D = R.D AND S.E=R.E] (** **CARTESIAN( R, S)** **))**

**לאחר הפעלת כלל 11b נקבל:**

**PI[R.D,S.E](NJOIN(R, S)** **)**

להלן רשימת האופרטורים:

**PI[...]( . )**

**SIGMA[...]( . )**

**CARTESIAN( . , . )**

**NJOIN( . , . )**

## חלק 2 - ארבעה Logical Query Plans שונים באופן רנדומלי

1. עליכם להפעיל את הכללים על 4 **עותקים שונים** של הביטוי האלגברי **מספר של כ - 10 איטרציות**.

2. אחרי שלב 1 אתם אמורים להיות במצב בו יש לכם לפחות **4 תכניות שונות** (Logical Query Plans) המיוצגות ע"י ביטויים אלגבריים שונים. הפלט שלכם אמור להציג את כל האיטרציות שביצעתם (למשל – "השתמשנו בכלל 4b וקיבלנו את הביטוי הבא...") וכמובן את 4 הביטויים הסופיים שאמורים להיות שונים.

## חלק 3 – Size Estimation

**קלט** – ארבעה Logical Query Plans שונים באופן רנדומלי שחושבו בחלק 2.

**פלט** – עליכם לפרק את הביטוי בכל אחת מה - 4 LQPs שלב אחר שלב (לפי הסוגריים) ובכל שלב לכתוב מהו ה - R (רוחב השורה) וה -n (מספר השורות) לפני ביצוע הפעולה ואחרי ביצוע הפעולה, על פי הכללים שנלמדו בכיתה ומופיעים במצגות ה - Size Estimation.

למשל: אם ניקח את ה LQP הבא:

**PI[R.D,S.E]( SIGMA[S.D=4 AND R.A=10] (** **CARTESIAN( R, S)** **))**

אז בשלב הראשון, נפנה לסוגריים הפנימיים ביותר (אלו של המכפלה הקרטזית) ונוציא פלט: מהו גודלם המקורי של R ו S (מבחינת רוחב שורה ומבחינת מספר השורות בכל אחת מהטבלאות) ונציין את רוחב השורה ואת מספר השורות של הטבלה המתקבלת לאחר הפעולה הקרטזית. לאחר שקיבלנו את רוחב השורה ומספר השורות של תוצאת המכפלה הקרטזית, נתונים אלה ישמשו כקלט על מנת להעריך את השפעתה של הפעולה הבאה (בדוגמא שניתנה כאן - פעולת ה SIGMA. תוצאת ה SIGMA תשמש כקלט להערכת השפעת ה PI).

כך יש לבצע לכל אחד מ - 4 ה - LQPs שקיבלנו כקלט לתרגיל.

את הנתונים לגבי הטבלאות עצמן (מספר השורות ורוחב הטבלה, נתונים על כל עמודה וכד') יש להחזיק בקובץ שניתן לעריכה.

להן מבנה הקובץ:

---------------------------------------------------------*--****statistics.txt****----------*------------------------------------------

Scheme R

R(A:INTEGER,B:INTEGER,C:INTEGER,D:INTEGER,E:INTEGER)

n\_R=100

V(A)=50

V(B)=100

V(C)=50

V(D)=1

V(E)=8

Scheme S

S(D:INTEGER,E:INTEGER,F:INTEGER,H:INTEGER,I:INTEGER)

n\_S=200

V(D)=100

V(E)=100

V(F)=100

V(H)=8

V(I)=200

------------------------------------------------------------------------------------------------------------------------------

לדוגמא הפלט יראה כך:

**CARTESIAN**

**input: n\_R=100 n\_S=200 R\_R=20 R\_S=20**

**output: n\_Scheme1=20000 R\_Scheme1=40**

**SIGMA[S.D=4 AND R.A=10]**

**input: n\_Scheme1=20000 n\_Scheme1=40**

**output: n\_Scheme2=4 R\_Scheme2=40**

**PI[R.D,S.E]**

**input: n\_Scheme2=4 n\_Scheme2=40**

**output: n\_Scheme3=4 R\_Scheme3=8**

**הערות:**

1. ניתן להניח שהאופרטורים היחידים שמשתתפים בתנאים הם: **AND =,**.
2. ניתן להניח שערכי ה -V (כלומר: מספר הערכים השונים בכל עמודה) נשארים ללא שינוי לאורך כל הסעיף. כמו כן ניתן להניח שהמידע מתפלג באופן אחיד.

**הסבר:**

עלינו לנתח את אופרטור הבחירה על סכמה בודדת, או על מכפלה קרטזית של שתי סכמות כמו בדוגמא למעלה.

יש 200 שורות בטבלה S, בעמודה D יש 100 שונים. כלומר: אם המידע מתפלג אחיד אז כל איבר יופיע בהסתברות של 1/100. בפרט כל איבר יופיע בעמודה S.D פעמיים בממוצע.

יש 100 שורות בטבלה R, בעמודה A יש 50 מספרים שונים. כלומר: אם המידע מתפלג אחיד אז כל איבר יופיע בהסתברות של 1/50. בפרט כל איבר יופיע בעמודה R.A פעמיים בממוצע.

במכפלה הקרטזית יש 20000 אלף שורות, וכל ערך של S.D מופיע עם כל ערך של R.A. לכן אם ההסתברות לראות את הערך 4 עבור S.D היא 1/100 וההסתברות לראות את הערך 10 עבור R.A היא 1/50 אזי ההסתברות לראות את הערך 10 וגם 4 היא 1/100\*1/50= 1/5000 .

בפרט השאילתה תחזיר 4 שורות מתוך 20000 השורות של המכפלה הקרטזית.

