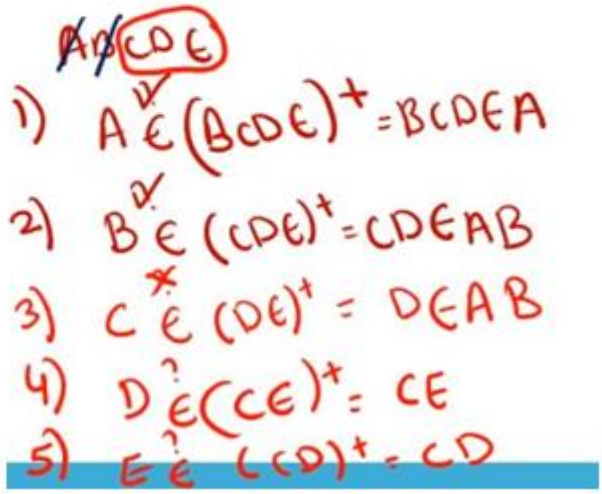
אלגוריתמים למסדי נתונים

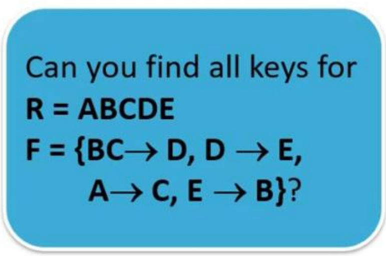
מציאת Superkeys

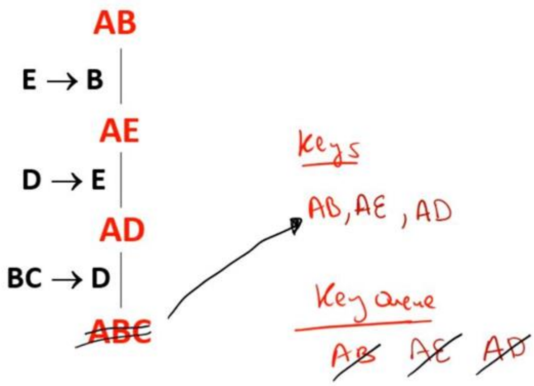
לוקחים את R ובכל פעם מנסים להוריד מישהו ולראות אם מקבלים אותו דרך הסגור של השאה, אם כן – מוחקים. אם לא – משאירים.



מציאת כל המפתחות

מתחילים מלמצוא מפתח אחד (בכל דרך שנבחר). שמים אותו בתור המפתחות, ואז שמים אותו בראש העץ ומוחקים מהתור. מחפשים תלות ב-F שבה צד ימין יש לו חיתוך לא ריק עם המפתח שלנו. מחליפים את כל מה שבצד שמאל עם צד ימין וממשיכים עד שלא יוצרים מפתח חדש. **לשים לב** – חשוב לעשות minimize לתהליך של יצירת מפתח חדש. (בדוגמה למטה קיבלנו ABC אבל אחרי minimize זה AB שכבר ראינו)





בדיקה האם הפירוק הוא ללא אובדן

עבור 2 יחסים

A decomposition of R into R_1,R_2 is a **lossless join decomposition** with respect to F if, at least one of the following holds:

1) $R_1 \subseteq (R_1 \cap R_2)^+$
2) $R_2 \subseteq (R_1 \cap R_2)^+$

פירוק של R לשני יחסים הוא ללא אובדן אם לפחות אחד מהם R_1, R_2 מוכל בסגור של חיתוכם.

Is this a lossless join decomposition of R = (A,B,C) when F = {A → B, B → C} ?

R_1 = (A,B)
R_2 = (B,C)

$$(R_1 \cap R_2)^+ = (B)^+ = BC$$

מתקיים $R_2 \subseteq BC$ ולכן הפירוק ללא אובדן.

עבור n יחסים

נחשב תקין כל עוד איחודו כולל את כל מי שהיה ב-R המקורי. איך בודקים? בונים טבלה שהעמודות זה כל האטריביוטים והשורות זה היחסים. אם האטריביוט נמצא בשורה, נשים a_j כאשר j מספר העמודה. אחרת נשים $b_{i,j}$ כאשר i מספר שורה j מספר עמודה.

R = (A,B,C,D)
R_1 = (A,B)
R_2 = (B,C)
R_3 = (C,D)

	A	B	C	D
AB	a_1	a_2	$b_{1,3}$	$b_{1,4}$
BC	$b_{2,1}$	a_2	a_3	$b_{2,4}$
CD	$b_{3,1}$	$b_{3,2}$	a_3	a_4

אחר כך נסתכל על קבוצת התלויות F ונתחיל לגרור לפיה. תמיד נתעדף לשים a על פני b, ואם יש שני b אז בוחרים שרירותית. ידוע למשל $B \rightarrow C$ אז:

	A	B	C	D
AB	a_1	a_2	$b_{1,3}$	$b_{1,4}$
BC	$b_{2,1}$	a_2	a_3	$b_{2,4}$
CD	$b_{3,1}$	$b_{3,2}$	a_3	a_4

	A	B	C	D
AB	a_1	a_2	a_3	$b_{1,4}$
BC	$b_{2,1}$	a_2	a_3	$b_{2,4}$
CD	$b_{3,1}$	$b_{3,2}$	a_3	a_4

אם סיימנו לטפל בסתירות והגענו למצב שיש שורה שבה יש רק ערכי a הפירוק ללא אובדן. אחרת הוא עם אובדן. נשים לב שאם הטבלה לא מכילה שורה עם a בלבד, אז יצרנו דוגמה נגדית.

חשוב – גם אחרי שטיפלנו בתלות כלשהי, כאשר ממשיכים לתלויות אחרות זה יכול ליצור בה סתירה מחדש.

שימור תלויות

כדי לבדוק אם פירוק משמר תלויות מספיק לבדוק שהתלויות שמוגדרות על **תתי היחסים** מתקיימות.

הטלה של F על R_i זה אוסף תלויות פונקציונאליות שנובעות מ-F וקשורות רק לאטריביוטים ב-R_i.

מתי פירוק משמר תלויות? כאשר כל $X \to Y$ שנובע מ-F נובע גם מאיחוד ההטלות של F על תתי היחסים (וההפך). הערה: מספיק לבדוק את התלויות ב-F עצמה ולא צריך את מה שנובע ממנה.

האלגוריתם:

נעבור תלות תלות ב-F. אם כל האטריביוטים ב-$X \to Y$ מוכלים יחד באותה תת-סכימה R_i אז אין מה לבדוק כי התלות בוודאי נשמרת. הבעיה היא בתלויות פונקציונאליות שלא מוכלות בתוך תת סכימה אחת:

$F = \{A \to B,\ B \to C,\ C \to D,\ D \to A\}$

$R = (A,B,C,D)$

$R_1=(A,B)$

$R_2=(B,C)$

$R_3=(C,D)$

על כל תלות שצריך לבדוק נסתכל על $Z = Z \cup ((Z \cap R_i)^+ \cap R_i)$. מאתחלים את $Z \coloneqq X$, במקרה הזה להיות D, ואז בוחרים תת-סכימה כלשהי ומנסים להגדיל את Z. אם הצלחנו לכלול את כל Y בתוך Z סיימנו והתלות נשמרת. **חשוב –** גם אם כבר כללתי R_j כלשהי, אחרי שהמצב ב-Z השתנה אותה R_j עשויה להגדיל את Z פעם נוספת:

$Z = D$

$R_1 \quad Z = D \cup ((\overbrace{D \cap AB}^{\phi})^+ \cap AB) = D$

$R_3 \quad Z = D \cup ((\underbrace{D \cap DC}_{D})^+ \cap DC) = DC$

$DABC$

$R_2 \quad Z = DC \cup ((\underbrace{DC \cap BC}_{C^+ = CDAB})^+ \cap BC) = DCB$

$R_1 \quad Z = DCB \cup ((\underbrace{DCB \cap AB}_{B^+ = BCDA})^+ \cap AB) = DCBA$

מציאת הצורה הנורמאלית של פירוק

למקרה שבו **נתון לנו** פירוק ואנחנו רוצים לבדוק מה הצורה הנורמאלית שלו.

כדי לעשות את זה צריך גם למצוא את התלויות הפונקציונאליות שמוגדרות על ההטלה (ואין לנו את זה).

$F = \{AB \rightarrow C, C \rightarrow A, C \rightarrow D\}$

$R = (A,B,C,D)$
$R_1=(A,C,D)$
$R_2=(B,C)$

האלגוריתם: (דוגמה עבור מציאת תלויות מעל R_1)

<u>שלב ראשון – תלות לכל תת קבוצה ב-R_1</u>:

$A \rightarrow$ $\quad$ $AC \rightarrow$ $\quad$ $ACD \rightarrow$

$C \rightarrow$ $\quad$ $AD \rightarrow$

$D \rightarrow$ $\quad$ $CD \rightarrow$

<u>שלב שני – חישוב הסגור שלה בחיתוך עם R_1</u>:

$A \rightarrow (A^+ \cap ACD) = A$ $\quad$ $AC \rightarrow ACD$ $\quad$ $ACD \rightarrow ACD$

$C \rightarrow CAD$ $\quad$ $AD \rightarrow AD$

$D \rightarrow D$ $\quad$ $CD \rightarrow CDA$

<u>שלב שלישי – בדיקת כל אחד מהיחסים לפי תנאי *3NF* או *BCNF*</u>:

$\checkmark\ A \rightarrow (A^+ \cap ACD) = A$ $\quad$ $\checkmark\ AC \rightarrow ACD$ $\quad$ $\checkmark\ ACD \rightarrow ACD$

$\checkmark\ C \rightarrow CAD$ $\quad$ $\checkmark\ AD \rightarrow AD$

$\checkmark\ D \rightarrow D$ $\quad$ $\checkmark\ CD \rightarrow CDA$

מציאת כיסוי מינימאלי

אנחנו **מקבלים קבוצת תלויות** ורוצים להישאר רק עם אלו שהכרחיות ולא נובעות מאחרות.

על כיסוי מינימלי לקיים:

1. צד שמאל אטריביוט בודד
2. שקול ל-*F*
3. לא ניתן להקטין יותר

Example: F = {A → B, ABCD → E, EJ → GH, ACDJ → EG}

האלגוריתם:

<u>שלב ראשון</u>: הפרדת כל התלויות כך שבצד ימין יש אטריביוט אחד

$A \to B$
$ABCD \to E$
$EJ \to G$
$EJ \to H$
$ACDJ \to E$
$ACDJ \to G$

<u>שלב שני</u>: הורדת אטריביוטים מיותרים מצד שמאל

עוברים על כל אחת מהתלויות ובודקים את הסגור אחרי ההורדה. אות שהורדנו לא מחזירים.

$E \overset{?}{\notin} BCD^+ = BCD$
$E \overset{?}{\in} ACD^+ = ACDBE$ ✓
$E \overset{?}{\notin} AD^+ = ADB$
$E \overset{?}{\in} AC^+ = ACB$

הורדנו את *B*, בסוף השלב נקבל כאן:

$A \to B$
$ACD \to E$
$EJ \to G$
$EJ \to H$
$ACD \to E$
$ACDJ \to G$

<u>שלב שלישי</u>: הורדת תלויות מיותרות (אם קיימת תלות שנובעת מהאחרות, נוריד אותה)

$ACD \to E$ מופיעה פעמיים, לכן בהכרח מיותרת פעם אחת.

האם *G* נמצא בסגור של *EJ* ללא התלות $EJ \to G$? לא. לכן אינה מיותרת.

האם *H* נמצא בסגור של *EJ* ללא התלות $EJ \to H$? לא. לכן אינה מיותרת.

$ACD \to E$ זו הפעם היחידה ש-*E* מופיע בצד ימין ולכן אינה מיותרת.

לבסוף נשים לב ש-$G \in ACDJ^+$ גם ללא $ACDJ \to G$ ולכן היא מיותרת.

כיסוי מינימלי:

✓ $A \to B$
$\cancel{ACD \to E}$
✓ $EJ \to G$
✓ $EJ \to H$
✓ $ACD \to E$
$\cancel{ACDJ \to G}$

מציאת פירוק 3NF

Find a 3NF decomposition for R = (A,B,C,D,E,G,H,J)
F = {A → B, ABCD → E, EJ → GH, ACDJ → EG}

האלגוריתם:

<u>שלב ראשון</u> – חישוב כיסוי מינימלי ל-F (עם אלגוריתם כיסוי מינימלי)

$A \to B$
$EJ \to G$
$EJ \to H$
$ACD \to E$

<u>שלב שני</u> – לכל תלות בכיסוי המינימלי $X \to A$ נוסיף לפירוק את הסכמה XA

$R_1 = AB$
$R_2 = EJG$
$R_3 = EJH$
$R_4 = ACDE$

<u>שלב שלישי</u> – בדיקה האם אחת מתתי הסכמות היא מפתח על. אם לא – נוסיף סכמה שהיא מפתח (כלשהו)

$R_1^+ = AB$
$R_2^+ = EJGH$
$R_3^+ = EJHG$
$R_4^+ = ACDEB$

אף אחת מהן לא מפתח, לכן נוסיף $R_5 = ACDJ$

<u>שלב רביעי</u> – בדיקה האם יש תת-סכמה שמוכלת בתת-סכמה אחרת

(בדוגמה אין כזאת)

מציאת פירוק BCNF

R = (A,B,C,D, E),
F= {AB → C,
DE → C,
B → E}

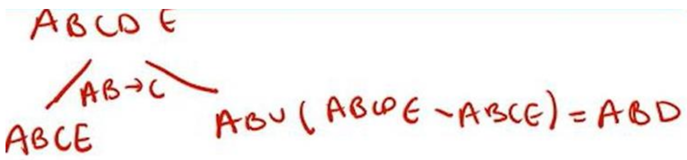
האלגוריתם:

שלב מקדים – אם $R \in BCNF$ נחזיר את R. (לא מתקיים בדוגמה)

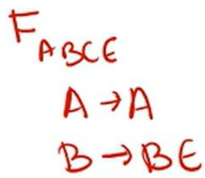
<u>שלב ראשון</u> – נחפש תלות $X \rightarrow Y$ שסותרת את תנאי ה-*BCNF*

בודקים למשל את הסגור של AB ורואים $AB^{+} = ABCE$

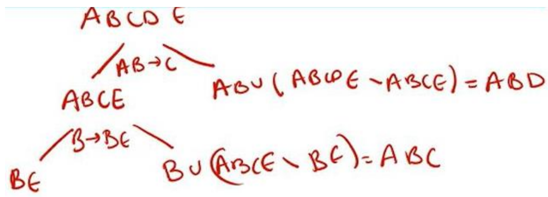
<u>שלב שני</u> – מפרקים את R לפי הסגור של AB בצד אחד, ו-AB איחוד מה שלא בסגור של AB בצד שני



<u>שלב שלישי</u> – עבור כל אחד מתתי היחסים, מוצאים את ההטלה וקבוצת התלויות הפונקציונליות. חוזרים לשלב ראשון.



כאן התלות $B \rightarrow BE$ מפרה את התנאי, מפרקים לפיה:



חוזרים עד שכל תתי-היחסים מקיימים את תנאי ה-*BCNF*.