**Ex4**

**ליאור שפירא וכרמל גרוס**

**שאלה 1:**

נניח:

* השדות הנומריים movieId, rating, year, duration, actorId, byear, dyear תופסים 4 בייט
* השדות הטקסטואליים character, name, genre, titleתופסים 10 בייט
* בטבלהMovies יש 10,000 שורות
* בטבלה Actors יש 50,000 שורות
* בטבלה PlaysInיש 100,000 שורות
* גודל בלוק הוא 8192 בייט.
* גודל החוצץ הוא בלוקים

נרצה לחשב את

1. ראשית נחשב את מספר הבלוקים שכל טבלה תופסת
   1. עבור block nested-loops נקבל ש-movies יהיה ביחס החיצוני כך שעלות החישוב הינה
   2. עבור hash join ראשית נחשב האם אפשר לבצע את האלגוריתם

אזי אפשר ולכן נקבל שעלות החישוב הינה

* 1. עבור sort-merge- join ראשית נחשב האם אפשר לבצע את האלגוריתם

אזי **לא ניתן לבצע את האלגוריתם**

1. כעת עבור
   1. עבור block nested-loops נקבל ש-movies עדין יהיה ביחס החיצוני כך שעלות החישוב הינה
   2. עבור hash join ראשית נחשב האם אפשר לבצע את האלגוריתם

אזי עדיין אפשר להפעיל את האלגוריתם ולכן נקבל שעלות החישוב הינה

* 1. עבור sort-merge- join ראשית נחשב האם אפשר לבצע את האלגוריתם

*אזי נוכל להפעיל את האלגוריתם ולמיין גם את אך לא בעלות האופטימלית שכן*

*אזי נקבל כי עלות החישוב הינה*

1. גודל החוצץ המינימלי עבור כל אלגוריתם-
   1. עבור block nested-loops נצטרך בלוק ל-movies בלוק ל-playsin ובלוק לoutput ולכן נצטרך מינימום **3 בלוקים**.
   2. עבור hash join נחשב את מספר הבלוקים המינימלי

אזי נצטרך לכל הפחות **9 בלוקים.**

* 1. עבור sort-merge- join בעלות לא אופטימלית ראינו בשאלה הקודמת כי עבור לא יכולנו לחשב אך עבור כן ולכן נניח כי נצטרך מינימום **16 בלוקים**.
  2. עבור sort-merge- join בעלות אופטימלית נצטרך

אזי נצטרך לכל הפחות **17 בלוקים.**

**שאלה 2:**

נרצה לחשב את הביטוי

גודל היחסים הם בכל בלוק של R יש 100 שורות וב-S 50.

ליחס S שני אינדקסים עם עלות גישה נמוכה על C,B וידוע B מפתח ביחס S.

1. גודל התוצאה בבלוקים של הוא

ראשית נחשב את מס השורות בטבלה- .

כמו כן נניח כי ההתפלגות אחידה (שכן אין מידע אחר) אזי נקבל כי מספר השורות המתאימות לתנאי הן - .

לבסוף נתון כי בכל בלוק 50 שורות ולכן נקבל כי מספר הבלוקים הינו

1. גודל התוצאה בבלוקים של הוא

שוב נחשב את מספר השורות בטבלה-

בהתאם לכלל אצבע שלמדנו בכיתה נניח כי מספר השורות המתאימות הוא שליש ולכן , כעת נתון כי בכל בלוק 100 שורות ולכן נקבל כי מס הבלוקים

1. מספר השורות בביטוי כולו הינו כפי שראינו בשיעור מתואר ע"י הנוסחה

כאשר מכיוון ש-B מפתח נקבל כי כמספר השורות

1. כעת נתאר את השאילתה לפי חלקים

ראשית נחשב האם כדאי לעשות full table scan או אינדקס לכל אחת משתי הselections.

מסעיפים קודמים ידוע כי

ב-R אין אינדקס ולכן כלומר full table scan

ב-S נעשה אינדקס על C ונראה כי העלות הינה

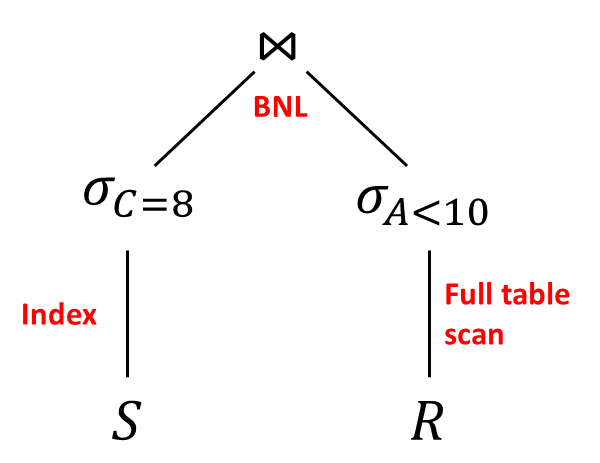
וללא אינדקס העלות תהיה

אזי נקרא את בעזרת אינדקס ואת כ-full table scan.

כעת נחשב את עלות החישוב של כל אלגוריתם עבור ה-join :

אזי SNJ לא יכול לפעול.

אזי נקבל כי **האלגוריתם הכי יעיל הינו BNL**

****

1. עלות החישוב היעילה ביותר הינה כפי שחושב בסעיף קודם.

**שאלה 3:**

נחשב את הביטוי

גודל היחסים הם גודל כל אטריביוט הוא 10 בייט וכל בלוק 2000 בייט. אין אינדקסים ואסור לבנות אותם, A מפתח של R.

1. *ראשית נחשב כמה שורות יש לכל תת שאילתה*

*עבור : מספר השורות הכולל ל-R הינו*

*עבור תנאי הבחירה נניח התפלגות אחידה על פני 10 ערכים אזי נקבל*

*עבור מספר השורות הכולל הינו*

*נניח התפלגות של שליש ונקבל*

*אזי מספר השורות בתוצאה (כאשר כי A מפתח ב-R)*

1. *נעבור על A,D ולכן מספר השורות בבלוק הוא אזי נקבל כי גודל התוצאה בבלוקים הוא* ***.***
2. כעת נתאר את השאילתה לפי חלקים

ראשית מכיוון שאין אינדקסים נוכל להגיד כי כל תת שאילתה של selection הינה full table scan.

אזי נקרא את ואת כ-full table scan.

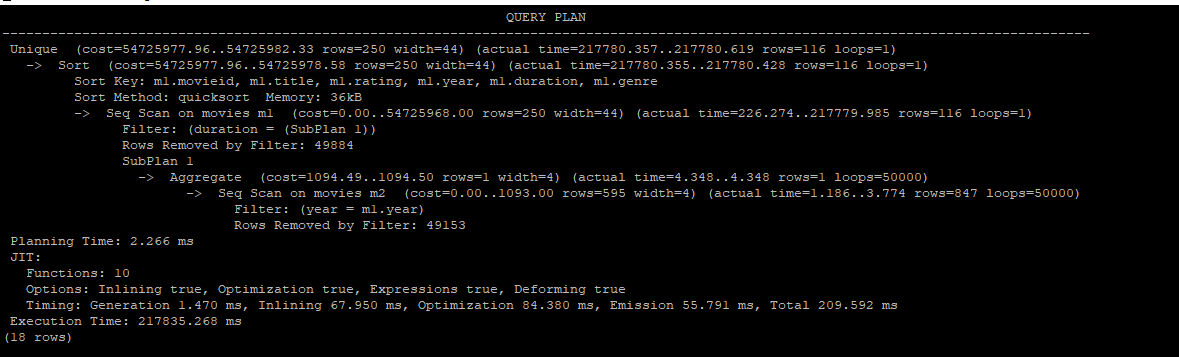
כעת נחשב את עלות החישוב של כל אלגוריתם עבור ה-join :

אזי קיבלנו כי פועלים בצורה הכי טובה באופן שווה



1. עלות החישוב היעילה ביותר הינה כפי שחושב בסעיף קודם.

**שאלה 4:**

1. השאילתה רצה ביותר משתי דקות
2. השאילתה החדשה היא-

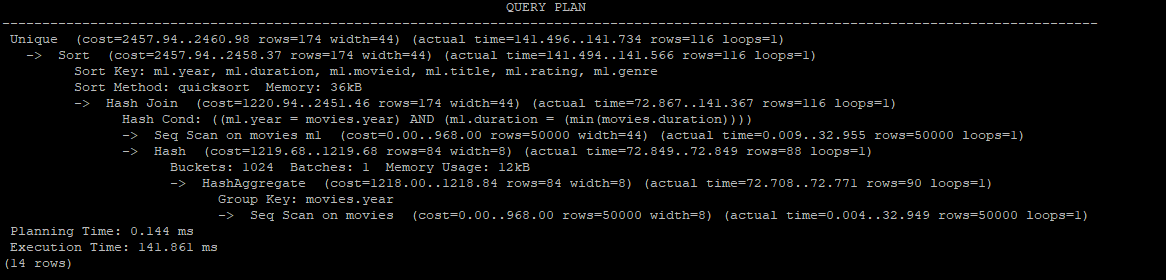
SELECT distinct \*

FROM Movies Natural join (SELECT year, min(duration) as dur FROM Movies

GROUP BY year) ;M2

זמן הרצת השאילתה הוא

להבנתנו, מה שגורם לשיפור בזמן הריצה הוא הימנעות ממעבר על Movies לכל סרט. כך, נעבור על Movies פעם אחת, כאשר קיבצנו את סרטים שיצאו באותה השנה, בחרנו את אורך הסרט המינימלי של כל קבוצה (אותו חישבנו פעם אחת), והשתמשנו בתוצאה על מנת להחזיר את הסרטים הרלונטים באמצעות natural join.

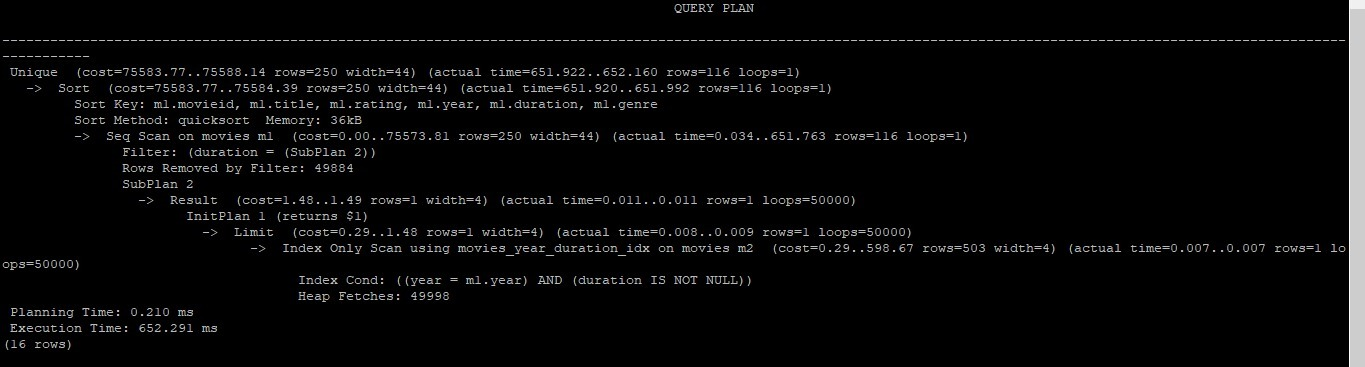


1. אפשר לשפר את זמן הריצה ע"י הוספת אינדקס.

ניסינו להריץ את השאילתה עם כל מיני אינדקסים למשל-

אינדקס על אינדקס על או על שני השדות.

נראה כי האינדקס המוצלח יותר היה זאת, משום שהוא שדה משותף בצירוף הטבעי וגם בבחירת משך הסרט המינימלי האינדקס יכול לעזור במקום לעבור אחד אחד על כל משכי הסרטים, וכך אנחנו יכולים לדעת איפה הערך המינימלי יהיה בעץ האינדקס ולגשת אליו בזמן קבוע.



לאחר הוספת האינדקס זמן הריצה הכולל היה

אפשר לראות כי לפני הוספת האינדקס המערכת בחרה להשתמש באלגוריתם לביצוע השאילתה ולאחר הוספת האינדקס המערכת בחרה באלגוריתם שכן הוספת האינדקס אפשרה את השימוש בו ועלותו הייתה נמוכה יותר.

בנוסף מכיוון שהגדרנו את האינדקס על 2 שדות, אין צורך לגשת לשורות ב-Movies אלא רק להגיע לעלים המתאימים בעץ.