מסדי נתונים (67506) | תרגיל 3

עופר פיינשטיין, 316413434 | נועה ברליאה, 318813789

26 בנובמבר 2020

חלק ב

שאלה 1

סעיף א

$$SELECT\ actorId$$

$$FROM\ PlaysIn$$

$$WHERE\ character\ = 'Sheriff'$$

סעיף ב

```
public=> select actorid from playsin where character='Sheriff';
public=> explain analyze select actorid from playsin where character='Sheriff';
                                          QUERY PLAN
-----------------------------------------------------------------------------------------------
 Seq Scan on playsin  (cost=0.00..615.15 rows=51 width=4) (actual time=0.593..3.295 rows=50 loops=1)
   Filter: (("character")::text = 'Sheriff'::text)
   Rows Removed by Filter: 32602
 Planning Time: 0.075 ms
 Execution Time: 3.351 ms
(5 rows)
```

זמן חישוב השאילתא הוא $PlanningTime + ExecutionTime$, כלומר $0.075 + 3.351 = 3.423\,ms$. מכיוון שחישוב השאילתא נעשה ללא אינדקסים על הטבלה, השאילתא מחושבת על ידי מעבר על כל שורות הטבלה, בדיקה האם שדה ה־$character'$ שווה לערך $Sheriff$, אם כן, השורה מתווספת לטבלה המוחזרת מהשאילתא.

סעיף ג

$$CREATE\ INDEX\ character\_idx\ ON\ PlaysIn(character)$$

סעיף ד

```
public=> create index character_idx on playsin(character);
CREATE INDEX
public=> explain analyze select actorid from playsin where character='Sheriff';
                                                    QUERY PLAN
-----------------------------------------------------------------------------------------------------------------------
 Bitmap Heap Scan on playsin  (cost=4.68..124.27 rows=51 width=4) (actual time=0.050..0.140 rows=50 loops=1)
   Recheck Cond: (("character")::text = 'Sheriff'::text)
   Heap Blocks: exact=37
   ->  Bitmap Index Scan on character_idx  (cost=0.00..4.67 rows=51 width=0) (actual time=0.040..0.040 rows=50 loops=1)
         Index Cond: (("character")::text = 'Sheriff'::text)
 Planning Time: 0.212 ms
 Execution Time: 0.190 ms
(7 rows)
```

כעת זמן חישוב השאילתא הוא $0.212 + 0.190 = 0.402\,ms$. מכיוון שהוספנו אינדקס על הטבלה לפי השדה $character$ אופן חישוב השאילתא נעשה בהתאם ל־$Index\,range\,scan$: השאילתא היא על ה־$search\,key$ ולכן חישוב השאילתא יהיה סכום עלות החיפוש בעץ (גובה העץ - הגעה לעלים) ומספר העלים המתאימים (המקיימים את התנאי על ה־$search\,key$).

שאלה 2

סעיף א

1. גודל כל בלוק 1000 בתים, וגודל כל שורה 150 בתים, לכן יש $\lfloor \frac{1000}{150} \rfloor = 6$ שורות בבלוק. בטבלה $Movies$ יש 10,000 שורות ולכן נצטרך $\lceil \frac{10000}{6} \rceil = 1,667$ בלוקים בזכרון. במקרה הגרוע ביותר נעבור על כל השורות בטבלה.

כלומר 1,667 פעולות I/O.

2. נחשב דרגת פיצול אופטימלית:

$$\begin{aligned} 8(d-1) + 8d &\leq 1,000 \\ 16d - 8 &\leq 1,000 \\ 16d &\leq 1,008 \\ d &\leq 63 \end{aligned}$$

גודל מצביע הוא 8 בתים, וגם גודל השדה $duration$ 8 בתים. בכל קודקוד יש לכל היותר d מצביעים, ולכל היותר $d-1$ ערכים, ולכן אי־השוויון למעלה חייב להתקיים. נרצה דרגת פיצול אופטימלית ולכן ניקח את ה־d המקסימלי.
כלומר דרגת הפיצול האופטימלית היא **63**.

3. שלב 1 - נחשב את עומק העץ:

$$\log_{\lceil \frac{63}{2} \rceil} 10,000 = \lceil \log_{32} 10,000 \rceil = 3$$

שלב 2 - נחשב את מספר העלים:

מכיוון שהשאילתה מחזירה "$exsists$", נרצה רק לדעת האם קיים ערך עם $duration > 100$ בעץ, לכן נבקר בעלה אחד.

בסה"כ נבצע $3 + 1$ פעולות, כלומר **4** פעולות I/O.

סעיף ב

1. גודל כל בלוק 1000 בתים, וגודל כל שורה 150 בתים, לכן יש $\lfloor \frac{1000}{150} \rfloor = 6$ שורות בבלוק. בטבלה $Movies$ יש 10,000 שורות ולכן נצטרך $\lceil \frac{10000}{6} \rceil = 1,667$ בלוקים בזכרון. במקרה הגרוע ביותר נעבור על כל השורות בטבלה.

כלומר **1,667** פעולות I/O.

2. נחשב דרגת פיצול אופטימלית:

$$\begin{aligned} 8(d-1)+8d &\leq 1,000 \\ 16d-8 &\leq 1,000 \\ 16d &\leq 1,008 \\ d &\leq 63 \end{aligned}$$

גודל מצביע הוא 8 בתים, וגם גודל השדה $duration$ 8 בתים. בכל קודקוד יש לכל היותר d מצביעים, ולכל היותר $d-1$ ערכים, ולכן אי־השוויון למעלה חייב להתקיים. נרצה דרגת פיצול אופטימלית ולכן ניקח את ה־d המקסימלי.
כלומר דרגת הפיצול האופטימלית היא **63**.

3. שלב 1 - נחשב את עומק העץ:

$$\log_{\lceil \frac{63}{2} \rceil} 10,000 = \lceil \log_{32} 10,000 \rceil = 3$$

שלב 2 - נחשב את מספר העלים:

נתון שהערכים מתפלגים אחיד בטווח $[0,200]$, נחשב את מספר השורות המתאימות:

$$\lceil \frac{200-100}{200-0} \rceil \cdot 10,000 = \frac{1}{2} \cdot 10,000 = 5,000$$

אשר נכנסות ב־ $\lceil \frac{5,000}{32-1} \rceil = 162$ בלוקים.

בסה"כ נבצע $3+162$ פעולות, כלומר **165** פעולות I/O.

סעיף ג

1. גודל כל בלוק 1000 בתים, וגודל כל שורה 150 בתים, לכן יש $\lfloor \frac{1000}{150} \rfloor = 6$ שורות בבלוק. בטבלה $Movies$ יש 10,000 שורות ולכן נצטרך $\lceil \frac{10000}{6} \rceil = 1,667$ בלוקים בזכרון. במקרה הגרוע ביותר נעבור על כל השורות בטבלה.

כלומר **1,667** פעולות I/O.

2. נחשב דרגת פיצול אופטימלית:

$$\begin{aligned} 8(d-1)+8d &\leq 1,000 \\ 16d-8 &\leq 1,000 \\ 16d &\leq 1,008 \\ d &\leq 63 \end{aligned}$$

גודל מצביע הוא 8 בתים, וגם גודל השדה $movieId$ 8 בתים. בכל קודקוד יש לכל היותר d מצביעים, ולכל היותר $d-1$ ערכים, ולכן אי־השוויון למעלה חייב להתקיים. נרצה דרגת פיצול אופטימלית ולכן ניקח את ה־d המקסימלי.
כלומר דרגת הפיצול האופטימלית היא **63**.

3. שלב 1 - נחשב את עומק העץ:

$$\log_{\lceil \frac{63}{2} \rceil} 10,000 = \lceil \log_{32} 10,000 \rceil = 3$$

שלב 2 - נחשב את מספר העלים:

השדה $movieId$ הוא מפתח בטבלה $Movies$, לכן מתאימה לתנאי ה־$WHERE$ שורה אחת לכל היותר, לכן נבקר בעלה אחד.

שלב 3 - נחשב את מספר הגישות לטבלה:

שורה אחת מתאימה בטבלה, לכן ניגש לטבלה פעם אחת.

בסה"כ נבצע $3+1+1$ פעולות, כלומר **5** פעולות I/O.

סעיף ד

1. גודל כל בלוק 1000 בתים, וגודל כל שורה 150 בתים, לכן יש $\lfloor \frac{1000}{150} \rfloor = 6$ שורות בבלוק. בטבלה $Movies$ יש 10,000 שורות ולכן נצטרך $\lceil \frac{10000}{6} \rceil = 1,667$ בלוקים בזכרון.

כלומר **1,667** פעולות I/O.

2. נחשב דרגת פיצול אופטימלית:

$$\begin{aligned} 10(d-1)+8d &\leq 1,000 \\ 18d-10 &\leq 1,000 \\ 18d &\leq 1,010 \\ d &\leq 56 \end{aligned}$$

גודל מצביע הוא 8 בתים, וגודל השדה $genre$ 10 בתים. בכל קודקוד יש לכל היותר d מצביעים, ולכל היותר $d-1$ ערכים, ולכן אי־השוויון למעלה חייב להתקיים. נרצה דרגת פיצול אופטימלית ולכן ניקח את ה־d המקסימלי.
כלומר דרגת הפיצול האופטימלית היא **56**.

3. שלב 1 - נחשב את עומק העץ:

$$\log_{\lceil \frac{56}{2} \rceil} 10,000 = \lceil \log_{28} 10,000 \rceil = 3$$

שלב 2 - נחשב את מספר העלים:

נתון שהערכים מתפלגים אחיד ב־4 קטגוריות, נחשב את מספר השורות המתאימות:

$$\frac{1}{4} \cdot 10,000 = 2,500$$

אשר נכנסות ב־ $\lceil \frac{2,500}{28-1} \rceil = 93$ בלוקים.

שלב 3 - נחשב את מספר הגישות לטבלה:

נרצה לחשב את הממוצע של ה־$duration$ של השורות המתאימות, ולכן נרצה לגשת לכל השורות המתאימות לתנאי בטבלה, כלומר לגשת ל־ 2,500 שורות. מכיוון שמספר השורות הרלוונטיות בטבלה גדול יותר ממספר הבלוקים של הטבלה כולה, נקח את המינימום.

$$\min(1667, 2500) = 1,667$$

בסה"כ נבצע $3 + 93 + 1,667$ פעולות, כלומר **1,763** פעולות I/O.

סעיף ה

1. גודל כל בלוק 1000 בתים, וגודל כל שורה 150 בתים, לכן יש $\lfloor \frac{1000}{150} \rfloor = 6$ שורות בבלוק. בטבלה $Movies$ יש 10,000 שורות ולכן נצטרך $\lceil \frac{10000}{6} \rceil = 1,667$ בלוקים בזכרון.

כלומר **1,667** פעולות I/O.

2. נחשב דרגת פיצול אופטימלית:

$$\begin{aligned} 10(d-1) + 8(d-1) + 8d &\leq 1,000 \\ 26d - 18 &\leq 1,000 \\ 26d &\leq 1,018 \\ d &\leq 39 \end{aligned}$$

גודל מצביע הוא 8 בתים, גודל השדה $genre$ 10 בתים וגודל השדה $duration$ הוא 8 בתים. בכל קודקוד יש לכל היותר d מצביעים, ולכל היותר $d-1$ ערכים, ולכן אי־השוויון למעלה חייב להתקיים. נרצה דרגת פיצול אופטימלית ולכן ניקח את ה־d המקסימלי.

כלומר דרגת הפיצול האופטימלית היא **39**.

3. שלב 1 - נחשב את עומק העץ:

$$\log_{\lceil \frac{39}{2} \rceil} 10,000 = \lceil \log_{20} 10,000 \rceil = 4$$

שלב 2 - נחשב את מספר העלים:

נתון שהערכים מתפלגים אחיד ב־4 קטגוריות, נחשב את מספר השורות המתאימות:

$$\frac{1}{4} \cdot 10,000 = 2,500$$

אשר נכנסות ב־ $\lceil \frac{2,500}{20-1} \rceil = 132$ בלוקים.

שלב 3 - נחשב את מספר הגישות לטבלה:

נרצה לחשב את הממוצע של ה־$duration$ של השורות המתאימות, מכיוון שהערך שמור לנו באינדקס לא נצטרך לגשת לטבלה.

בסה"כ נבצע $4+132$ פעולות, כלומר **136** פעולות I/O