

מבחן LEVERAGING BIG DATA

סמסטר א

3 בפברואר 2014

המרצים: עידית כהן, טובה מילוא, עמוס פיאט, חיים קפלן

אין מגבלה לגבי חומר עזר

יש לפתור ארבע מתוך חמש השאלות, כל שאלה בעמוד אחד במחברת, כתוב תשובות קצרות מנוסחות היטב. אין צורך לחזור על הוכחות שניתנו בכיתה. כל שאלה 25 נקודות.

בהצלחה!

1. (25 נקודות) נתון האלגוריתם הבא, שפועל על stream של נקודות במרחב מטרי, המרחק בין כל זוג נקודות < 1 . $\text{dist}(T, q)$ הוא המרחק בין נקודה q לנקודה הכי קרובה בקבוצה T ל- q .

$T \leftarrow \text{emptyset}$

$R \leftarrow 1$

Repeat forever:

While $|T| \leq k$

do

Get next point p

If $\text{dist}(T, p) > R$ then add p to T

enddo

(α) All pairwise distances in $T > R$, $|T|=k+1$ points

$\text{Max dist}(T, q) < 2R$ for all q up to now

$R \leftarrow 2R$

$T' \leftarrow \text{emptyset}$

While (Exists q in T such that $(\text{dist}(q, T') > R$ or $T' = \text{emptyset})$) then add q to T'

$T \leftarrow T'$

End of Repeat

Let OPT be the minimum, over all k centers, of the maximal distance between the set of centers and the set of all points up to now.

א. הוכח כי הטענה המסומנת ב- α נכונה בכל פעם כשהלולאה לפני α נגמרת.

ב. הוכח כי ב- α : $R/2 < \text{OPT}$ ולכן $\text{Max}_q \text{dist}(T, q) < 4 \text{OPT}$.

ג. הוכח כי תמיד נכון ש $\text{Max}_q \text{dist}(T, q) < 8 \text{OPT}$.

2. (25 נקודות) ברשותינו DATA של יחס זוגות (סרט, אדם) WATCH המתאים לצפיות של אנשים בסרטים וכן יחס חברים של (אדם, אדם) FRIEND

ברצוננו לחשב לכל סרט y את $|G(y)|$ כאשר $G(y)$ היא קבוצת האנשים שצפו בסרט או הינם חברים של אדם שצפה בסרט. שים לב כי

$$G(y) = W(y) \cup \left(\bigcup_{x \in W(y)} F(x) \right)$$

כאשר $F(x)$ היא קבוצת החברים של אדם x ו- $W(y)$ היא קבוצת הצופים בסרט y .

- א. תארי/ כיצד לחשב את $|G(y)|$ לכל סרט y ע"י פלטפורמת MAPREDUCE הגדר/י את פונקציות ה- MAP וה- REDUCE בכל איטרציה.
- ב. בטא/י את מספר הזוגות (KEY,VALUE) הכולל המיוצר בחישוב בסעיף הקודם ואת המספר המכסימלי של הזוגות עם אותו KEY (הממופים לאותו reducer).
- ג. תארי/ כיצד תשתמש/י ב- **bottom-k Min-Hash sketches** בכדי לאמוד את $|G(y)|$ לכל y .
- ד. תארי/ כיצד לאמוד את $|G(y)|$ לכל סרט y ע"י **bottom-k Min-Hash sketches** על פלטפורמת MAPREDUCE.
- ה. בטא/י את מספר הזוגות (KEY,VALUE) הכולל המיוצר בחישוב בסעיף הקודם ואת המספר המכסימלי של הזוגות עם אותו KEY (הממופים לאותו reducer).

3. (25 נקודות) עליך לתכנן Linear sketch לצורך Exactly1?. כלומר ה-SKETCH מופעל על ווקטור b במימד n והפרדיקט Exactly1? הוא TRUE אם"ם לוקטור b בדיוק כניסה אחת שאינה 0.

ה-SKETCH הוא ווקטור במימד זוגי $2k$

לבחירתך שתי אפשרויות של פונקציות HASH (הנח/י שהן perfectly random)

A. k פונקציות $h_i: \{1, \dots, n\} \rightarrow \{0, 1\}$ לכל $i = 1, \dots, k$ ה-SKETCH יכול לטפל את $\sum_{j|h_i(j)=1} b_j$ ואת $\sum_{j|h_i(j)=0} b_j$.

B. פונקציית בודדה $g: \{1, \dots, n\} \rightarrow \{0, \dots, 2k-1\}$ וה-SKETCH יכול לטפל לכל $t = 0, \dots, 2k-1$ את $\sum_{j|g(j)=t} b_j$.

הנח/י שווקטור הקלט למשך כל העדכונים הוא כזה שכל הערכים אי-שליליים.

א. לכל אפשרות, תאר/י את המבנה של המטריצה הכופלת את ווקטור הקלט בכדי לקבל את ה-SKETCH המתאים.

ב. כיצד תעדכן/י את ה-SKETCH בהנתן עדכון מהצורה $b_i \leftarrow b_i + x$ לכניסה i של ווקטור הקלט?

ג. מהו מספר הפעולות ה-HASH ומספר הכניסות ב-SKETCH שעלינו לשנות בכל עדכון בכל אפשרות?

ד. כיצד תעריכ/י Exactly1? על סמך ה-SKETCH בכל אפשרות? מה ההסתברות לכל סוג של טעות (False positive/negative)?

ה. מה היתרון של אפשרות A על B ומה היתרון של אפשרות B על A?

4. **(25 נקודות)** נתון מסד נתונים המכיל n (הנח כי n מאד גדול) תמונות כל אחת היא מטריצה של 1024×1024 שורות ו- 1024 עמודות של פיקסלים כל אחד בן ביט אחד. נרצה לבנות מבנה נתונים שבהינתן תמונת שאילתה q נוכל למצוא בהסתברות $1/4$ תמונה באוסף השונה מ- q בלכל היותר 2048 ביטים במידה ויש באוסף תמונה השונה מ- q בלכל היותר 1024 ביטים. תאר (לפי הסעיפים הבאים) מבנה נתונים שעונה על שאילתה כזו בזמן $O(\sqrt{n})$ (הזנח גורמים לוגריתמים והשתמש בקרוב $\log(1-x) \cong -x$ כאשר x מאד קטן).
- א. תאר את המבנה עצמו כיצד נאחסן את המבנה בזכרון כך שלא יצרוך יותר מידי מקום? כמה מקום יתפוס המבנה?
 - ב. תאר את אלגוריתם השאילתה במדויק.
 - ג. נתח את זמן הריצה של אלגוריתם השאילתה.
 - ד. הוכח את נכונות האלגוריתם.

5. (25 נקודות) מוקצות על הדיסק שתי מטריצות A ו-C עם n שורות ו-n עמודות כאשר n גדול לפחות פי 100 מגודל הזיכרון M. המטריצה C ריקה בתחילה. המטריצות מאוכסנות על הדיסק "לפי שורות" כלומר ב-B/n הבלוקים הראשונים מתוך הבלוקים המכילים את המטריצה מופיעים האיברים שבשורה הראשונה ב-B/n הבלוקים הבאים נמצאים האיברים שבשורה השנייה וכו' (B) הוא גודל הבלוק או באופן מדויק יותר מספר האיברים מהמטריצה שנכנסים בבלוק). מריצים את האלגוריתם הבא על מנת ש-C תכיל את A^t (כלומר העמודה הראשונה של B תכיל את השורה הראשונה של A, העמודה השנייה את השורה השנייה וכו').

```
For r = 1 to n do
  For c = 1 to n do
    C[c,r] := A[r,c]
```

- האם האלגוריתם הוא cache oblivious? נמק.
- כמה קריאות של בלוקים לזיכרון וכתובות של בלוקים לדיסק יתבצעו במהלך הריצה? נמק.
- הנח כי $M \geq 3B^2$ תאר אלגוריתם פשוט שהוא IO-efficient (אבל לא בהכרח cache-oblivious) שמבצע את החישוב $C := A^t$ ב-B $O(n^2/B)$ פעולות I/O. (אין צורך לכתוב קוד, מספיק תאור מדויק במילים)
- האם ניתן לבצע את החישוב בפחות מ- $O(n^2/B)$ פעולות I/O? נמק.

