ב"ה

אלגוריתמים בחיפוש מידע – מועד א', תשע"ה – 89-529-01

כז' שבט תשע"ה, 16/02/2015

שעה 12:00

פרופ' עמיהוד אמיר

אין להשתמש בחומרי עזר

1. (33 נקודות) *בעיית החילופים המורחבת* (Generalized Swaps) הוגדרה בהרצאה. אנו עוסקים כרגע במקרה מיוחד של הבעיה – SGS – בה אורך התבנית שווה לאורך הטקסט והאלפבית הוא: $\sum=\{1,2,3\}$. באופן פורמלי, בעיית SGS מוגדרת כדלהלן:
קלט: טקסט $T=t_0,\ldots,t_m$, ותבנית $P=p_0,\ldots,p_m$ מעל אלפבית $\sum=\{1,2,3\}$.
הכרע: האם P מתאים ל T תחת swaps מורחב, כלומר מותר להחליף שני תוים שאינם סמוכים זה לזה אך אסור להזיז תו יותר מפעם אחת.

כתוב אלגוריתם הפותר את בעיית ה SGS. מה רעיון האלגוריתם שלך? מה זמן הריצה שלו? נמק. (בהנחה שהאלגוריתם נכון, ככל שהאלגוריתם יעיל יותר יינתנו יותר נקודות.)

<u>רעיון הפתרון:</u>

נגדיר תשע קבוצות: S<a,b> *עבור* a=1,2,3 *ו* b=1,2,3 *כאשר*
S<a,b> *מכיל את כל הזוגות* $<t_i,p_i>$ *עבורן* t_i=a *ו* p_i=b.
יש להוכיח או לנמק את טענת העזר הבאה:
***יש* swap *מורחב אמם לכל* a,b |S<a,b>| = |S<b,a>|** .
בהנחה שטענה זאת מוכחת, קל להראות שניתן לפתור את הבעיה בזמן לינארי.

קודי שגיאה:

1. אלגוריתם ללא הסבר הרעיון: 5-
2. נאיבי $O(m \log m)$ הסתברותי : 13-
3. טוען שמספר פונקציות h קבוע כי הא"ב קבוע (מספר הפונקציות לא קבוע אפילו לא"ב בינרי לפי ההוכחה שראינו): 6-
4. נאיבי $O(n^2)$ דטרמיניסטי : 17-
5. חיפוש במערך ממויין בזמן $O(1)$. מדוע ואיך?: 5-
6. הסתברותי $O(m)$: 7-

2. (33 נקודות) באלגוריתם ה LCA של ברקמן (שראינו בכתה) נבנה עץ בינארי מלא מעל מערך המספרים הנתון, כשבכל קדקוד בעץ רשומים המינימומים של כל הרישות של שרשור העלים שלו משמאל לימין, והמינימומים של כל הסיפות.

כתוב אלגוריתם הבונה את העץ. מה רעיון האלגוריתם שלך? מה זמן הריצה שלו? נמק. (בהנחה שהאלגוריתם נכון, ככל שהאלגוריתם יעיל יותר יינתנו יותר נקודות.)

<u>רעיון הפתרון:</u>

יש להראות איך כל קודקוד בעץ יכול לבנות את רשימת הרישות ורשימת הסיפות בזמן לינארי באורך הרשימות של בניו. היות וס"ה הרשימות צורכות מקום $O(n \log n)$ *הרי שהתשובה מוכחת.*

קודי שגיאה:

1. אלגוריתם נכון, זמן $O(n)$: 7-
2. אלגוריתם נאיבי : 16-
3. אין פרטים: 25-
4. פרטים לטיפול בסיפות חסרים: 5-

3. (34 נקודות) <u>*בעיית ה BM*</u> (Block Mass) מוגדרת כדלהלן:

פלט: כל המקומות i בטקסט עבורם קיימים $k_0, \dots, k_m$ כך ש:

$$\sum_{j=0}^{k_0-1} t_{i+j} = p_0$$

$$\sum_{j=0}^{k_1-1} t_{i+k_0+j} = p_1$$

.
.
.

$$\sum_{j=0}^{k_m-1} t_{i+k_0+k_1+\dots+k_{m-1}+j} = p_m$$

דוגמה: T=1,2,3,1,1,2,1,1,3,1,2,1 ו P=5,4.
אז באינדקס 1 יש הופעה של P כי 5=2+3 וגם 4=1+1+2.
(ראה מלבנים :T=1,2,3,1,1,2,1,1,3,1,2,1). גם באינדקסים 2,3,4 ו 6 יש הופעות.

כתוב אלגוריתם הפותר את בעיית ה BM. מה רעיון האלגוריתם שלך? מה זמן הריצה שלו? נמק. (בהנחה שהאלגוריתם נכון, ככל שהאלגוריתם יעיל יותר יינתנו יותר נקודות.)

<u>**רעיון הפתרון:**</u>

יש כמה דרכים לטפל בבעיה. הדרך הפשוטה ביותר היא לתרגם את הטקסט לאונרי (דבר שיגדיל את אורכו פי 3) בצורה הבאה: השאר את 1 ללא שינוי. כתוב את 2 בצורה: 10 ואת 3 בצורה 100. כלומר הטקסט בדוגמה יראה:
1101001110111001101

נשים לב שכל מקום במחרוזת החדשה בו יש 1, מקביל לאינדקס במערך המקורי!

כמו כן, יש הופעה במחרוזת הקלט במקום i אם יש 1-ים באינדקסים: j (המתאים לi במחרוזת החדשה) וכן בכל האינדקסים $j + \sum_{\ell=0}^{k} p_\ell$ לכל $k = 0, \dots, m$.

אם $m < \log n$ אזי הדרך הטובה ביותר לבדוק זאת היא בזמן $O(nm)$ אבל אם $m \geq \log n$
אזי הדרך המהירה ביותר לבדוק זאת היא ע"י FFT ואז הזמן הוא $O(n \log n)$ כלומר הזמן האופטימלי הוא: $O(n \min(m, \log n))$.

קודי שגיאה:

1. נאיבי $O(mn)$: 10-
2. עץ סיפות. בעיה: מעניין אותנו גם סכום שמסתיים באמצע קשת (כי אולי אפשר להמשיך משם) לכן הזמן הוא לא $O(n+tocc)$ אלא $O(n^2)$, כמו הנאיבי : 22-
3. בניית כל המחרוזות האפשריות מ P. טענה שגויה שיש רק $O(log\ m)$ כאלה. למעשה זה אקספוננציאלי. 24-
4. נאיבי $O(n^2)$: 19-
5. נאיבי $O(n^3)$: 24-
6. נאיבי $O(n^2)$ אבל טוען שזה $O(mn)$: 24-
7. נאיבי $O(n^2)$ אך טוען שזה $O(n)$: 29-
8. $O(n^2)$ אך טוען שזה $O(n\sqrt{m}\log m)$: 27-
9. נאיבי $O(nm)$ אך טוען ללא הוכחה שהממוצע $O(n)$: 19-
10. נאיבי $O(n^2)$ לבניית כל הסכומים בטקסט ואח"כ *FFT* שלא יעבוד : 24-
11. רעיון עם מצביעים שאמור לתת זמן $O(mn)$ אך לא עובד: 15-
12. נאיבי $O(nm)$ אך טוען שזה $O(n)$: 20-

בהצלחה