מבני נתונים
פתרון תרגיל 5

גלעד אשרוב

11 במאי 2012

שאלה 1.

א. תאר אלגוריתם המקבל כקלט רשימה מקושרת ממויינת ומחזיר רשימת דילוגים אידיאלית. מה זמן הריצה של האלגוריתם שהצגת?

פתרון: האלגוריתם פשוט עובר בלולאה, ומעלה כל איבר שני "רמה". כלומר, לוקחים את הרשימה התחתונה (הרשימה המקורית), ומעלים כל איבר שני. לאחר שבנינו את הרמה השנייה, אנחנו מפעילים שוב את האלגוריתם שהצגנו כעת על רמה זו (כלומר, שוב מעלים כל איבר שני רמה). מסיימים לאחר $\log n$ רמות. סה"כ זמן ריצה:

$$n + n/2 + n/4 + n/8 + \ldots + 1 \leq 2n$$

כלומר, הלאוגירתם רץ בזמן $O(n)$.

ב. מה ההסתברות שבהכנסת n איברים לרשימה, נגיע למצב שבו מספר הרמות מגיע ל - $\omega(\log n)$?

פתרון: ההסתברות שאיבר אחד מסויים עובר $\omega(\log n)$ רמות:

$$\Pr\left[x_i \text{ has promoted more than } \omega(\log n) \text{ times}\right] \leq 2^{-\omega(\log n)} = \frac{1}{2^{\omega(\log n)}} \approx \frac{1}{n^{\omega(1)}}$$

כפי שראינו בכיתה, בעזרת חסם האיחוד:

$$\Pr\left[\text{skip list has more than } \omega(\log n) \text{ levels}\right] \leq \frac{1}{n^{\omega(1)-1}}$$

בעצם, ההסתברות הנ"ל קטנה מ**כל פולינום**. כלומר, כל פונקציה שהיא ב $n^{\omega(1)-1}$ היא פונקציה **סופר פולינומיאלית** (גדולה בסדר גודל מכל פולינום), והערך $\frac{1}{n^{\omega(1)-1}}$ קטן מכל $1/p(n)$, לכל פולינום $p(n)$. (כמובן, שהכל באופן אסימפטוטי, כלומר, החל מ n_0 מסויים). להסתברות כזו (קטנה מכל פולינום) אנחנו קוראים **הסתברות זניחה**.

לסיכום, כפי שראינו בכיתה - ההסתברות שנהיה ב $O(\log n)$ רמות היא הסתברות **פולינומית** בעוד שההסתברות שנהיה ב - $\omega(\log n)$ רמות היא **הסתברות זניחה**.

ג. כדי להימנע ממצב שבו מספר הרמות גדל וגדל ללא הגבלה, סטודנט הציע את השיפור הבא: כאשר מספר הרמות גדל מ- $5\log n$, נעצור, ונבנה מחדש את הרשימה כרשית דילוגים אידיאלית. ברור כי זמן חיפוש הוא בתחולת $O(\log n)$, ומספר הרמות הוא לכל היותר $5\log n$. מהי תוחלת זמן ההכנסה?

פתרון: ההסתברות שנעלה מעל ל- $5\log n$ רמות היא $1/n^4$. במקרה זה, אנחנו עוצרים ובונים מחדש את הרשימה ($O(n)$). נקבל, תוחלת זמן ההכנסה קטנה שווה ל:

$$1/n^4 \cdot O(n) + (1 - 1/n^4) \cdot O(\log n) \leq O(\log n).$$

ד. הצג אלגוריתם לאיחוד שתי רשימות דילוגים. מהו זמן הריצה? הסבר מדוע הפתרון שלך הוא האופטימלי.

פתרון: נסתכל פשוט על הרמה התחתונה. נקח את שתי הרשימות, שתיהן ממויינות, ונמזג אותן לרשימה ממויינת אחת. אם ברשימה הראשונה n איברים, ובשנייה m איברים, המיזוג ייעלה לנו $O(n+m)$. לאחר מכן, נבנה רשימת דילוגים אידיאלית על הנתונים האלה (כמו בסעיף א') - בעלות $O(n+m)$. זהו הזמן האופטימלי, מכיוון שרק לקרוא את הנתונים עולה $O(n+m)$.

שאלה 2. פתרון יתפרסם בלי נדר בהמשך.

שאלה 3. נתבונן בשיטה הבאה למיון קבוצה A_0 למיון של n מספרים שונים: נחלק את A_0 ל-$\sqrt{n}$ תת־קבוצות זרות בגודל $\sqrt{n}$ כל אחת. אח"כ נבחר בכל פעם את האיבר הקטן ביותר מתוך אלה שעוד לא נבחרו ע"י שנמצא (בזמן לינארי) את האיבר הקטן ביותר בתוך כל אחת מ־ $\sqrt{n}$ הקבוצות (נקרא לקבוצות $\sqrt{n}$ הקטנים האלה A_1), ואז נמצא את האיבר הקטן ביותר ב־ A_1.

א. כמה זמן לוקח מציאת האיבר הקטן ביותר? והשני? ה־ ℓ, $\ell = 2, 3, \ldots, n$? סכם – מה הסיבוכיות של שיטת מיון זו?

ב. האם יעזור אם, במקום לחזור ולפש שוב ושוב את האיבר הקטן ביותר בכל קבוצה, נמיין את $\sqrt{n}$ הקבוצות מיד בהתחלה?

ג. כדי לשפר נחלק את A_0 ל - $n^{2/3}$ קבוצות בגודל $n^{1/3}$ כל אחת. נקבל קבוצה A_1 בגודל $n^{2/3}$ ולכן גם אותה נחלק ל - $n^{1/3}$ קבוצות בגודל $n^{1/3}$. נקרא לקבוצת האיברים המינימאליים בשם A_2. כמה זמן לוקח מציאת האיבר הקטן ביותר? והשני? ה־ ℓ, $\ell = 2, \ldots, n$? סכם: מהי הסיבוכיות של שיטת מיון זו?

ד. תאר בקצרה הכללה של השיטה של סעיף א' (רמה נוספת אחת) ושל סעיף ג (2 רמות נוספות) ל־ k רמות נוספות. מה תהיה הסיבוכיות?

ה. מהו הערך הגדול ביותר של k שהוא אפשרי בסעיף ג (רמז: חייבים להיות לפחות 2 איברים בכל קבוצה בחלוקה). מה תהיה הסיבוכיות במקרה זה?

פיתרון.

א. בכדי למצוא את האיבר הקטן ביותר – נמצא ראשית את המינימום בכל אחת מהקבוצות, ולאחר מכן נמצא את המינימום בקבוצת ה"נציגים" (המינימומים). מציאת מינימום בכל קבוצה – $\sqrt{n}$, ישנן $\sqrt{n}$ קבוצות, ולכן בניית מערך A_1 עולה $O(n)$ פעולות. כעת, לאחר בניית מערך זה, מציאת המינימום ב־ A_1 לוקחת גם היא $\sqrt{n}$. בסה"כ, מציאת המינימום לוקחת לנו $O(n+\sqrt{n})$. עם כל איבר שהכנסו ל - A_1 נשמור מאיזו קבוצה הוא הגיע.

האיבר המינימלי הבא (האיבר השני בגודלו בסה"כ) יכול להיות אחד מהשניים – או אחד מהאיברים שבקבוצה A_1, או איבר בקבוצה שממנה נלקח אותו האיבר המינימלי. לכן, לאחר שמצאנו את המינימום ב־ A_1 נלך לקבוצה המקורית ממנה נלקח האיבר הנ"ל, ונמצא את האיבר השני בגודלו בקבוצה (בזמן לינארי. ניתן לסמן את האיברים שהוצאנו כבר בקבוצה זו). את האיבר הנ"ל נציב ב־ A_1 במקום האיבר המינימלי שמצאנו בשלב הקודם.

עלות מציאת האיבר השני בגודלו כוללת למעשה שני חיפושי מינימום - הראשון חיפוש מינימום מאחת הקבוצות של A_0, השני מציאת המינימום של הקבוצה A_1. לכן בסה"כ, עלות מציאת האיבר השני בגודלו הינה $O(\sqrt{n})$.

נמצא את האיבר השלישי בגודלו בדיוק כמו שמצאנו את האיבר השני בגודלו (כלומר - עדכון A_1, ומציאת המינימום ב - A_1). לכן, עבור כל אחד מהאיברים האחרים נשלם $O(\sqrt{n})$. ובסה"כ:

$$T(sort) = O(n + n \cdot 2\sqrt{n}) = O(n\sqrt{n}) = O(n \cdot n^{1/2})$$

ב. מיון קבוצה בגודל $\sqrt{n}$ עולה $O(\sqrt{n}\log\sqrt{n})$. אם נמיין כל אחת מהקבוצות, נשלם $\sqrt{n} \cdot O(\sqrt{n}\log\sqrt{n}) = O(n\log\sqrt{n}) = O(n\log n)$ עבור בניית כל הקבוצות. כעת, מציאת המינימום הבא מכל קבוצה ייקח $O(1)$, אבל מציאת המינימום ב - A_1 תקח $O(\sqrt{n})$. אם כן, נשלם בסה"כ:

$$T(sort) = O(n\log\sqrt{n} + n \cdot \sqrt{n}) = O(n\sqrt{n}) = O(n \cdot n^{1/2})$$

כלומר, המיון לא שיפר את זמן הריצה.

ג. בכדי לבנות את A_1, נמצא $n^{2/3}$ מינימומים ב - A_0. מציאת כל מינימום כזה עולה $O(n^{1/3})$, ולכן בסה"כ נשלם $O(n)$. בכדי לבנות את A_2, נמצא $n^{1/3}$ מינימומים, כאשר מציאת כל מינימום כזה עולה $n^{1/3}$. לכן, בסה"כ נשלם $O(n + n^{2/3}) = O(n)$.

כעת יש לנו 3 רמות. בכל שלב, אנו מוצאים את המינמום של A_2 (בעלות $O(n^{1/3})$). כאשר נוציא את אותו המינימום, נחפש במקומו את האיבר המינימלי מהקבוצה ממנה הגיע ב - A_1 (שוב, עלות $O(n^{1/3})$). כאשר נוציא את האיבר המינימלי באותה הקבוצה ב - A_1, נחפש במקומו איבר מינימלי מ - A_0 (שוב $O(n^{1/3})$). בס"כ, מציאת כל איבר עולה - $O(3 \cdot n^{1/3})$. נשלם אם כן בסה"כ:

$$T(sort) = O(n + n \cdot 3 \cdot n^{1/3}) = O(n \cdot n^{1/3}).$$

ד. כל קבוצה תהיה בגודל $n^{1/k}$. כלומר, ב- A_k- תהיה קבוצה אחת בגודל $n^{1/k}$. כל איבר כזה מגיע כנציג מתוך קבוצה בגודל $n^{1/k}$ ב - A_{k-1}. כלומר, ב - A_{k-1} יהיו $n^{1/k}$ קבוצות בגודל $n^{1/k}$ כל אחת, ובסה"כ יהיו $n^{2/k}$ איברים מ - A_{k-1}. בצורה כזו נמשיך עד שנגיע ל - A_0, שיהיה מחולק ל - $n^{k-1/k}$ קבוצות בגודל $n^{1/k}$ כל אחת. בכדי למצוא את המינמום - נאלץ לבצע עבודה בעלות של $k \cdot n^{1/k}$. עלות הבניה תישאר $O(n)$, ועלות הכוללת למיון: $O(n \cdot k \cdot n^{1/k})$.

ה. k המקסימלי מתקבל כאשר $n^{1/k} = 2$. נקבל - $k = \log n$ (ראינו בתרגול למה זה נכון). נקבל:

$$T(sort) = O(n \cdot \log n \cdot n^{1/\log n}) = O(n \cdot \log n \cdot 2) = O(n\log n)$$

שאלה 4. תבנית : ccccba טקסט: ccccaabbbbb

מנתחים את התבנית ומתקיים $\Delta_1(a) = 0$, $\Delta_1(b) = 1$, $\Delta_1(c) = 2$.

בהצבה הראשונה של הטקסט מול התבנית מתחילים עם $i = 5$ (נניח שהאינדוקס מתחיל מ-0). האותיות שוות (a) ולכן עוברים ל- $i = 4$. שם יש השוואה בין b בתבנית ל-a בטקסט. הפעולה שעושים היא $i = i + \Delta_1(T(i))$ $T(2) = a$ כלומר מקדמים את i ב-0, ומתחילים לעשות השוואה נוספת בין התבנית (מהתו האחרון שלה), לטקסט, כשמתחילים מ-$i = 4$.

האותיות שוות (a) ולכן עוברים ל-$i = 3$. שם יש השוואה בין b בתבנית ל-c בטקסט. הפעולה שעושים היא $i = i + \Delta_1(T(i))$ $T(c) = 2$ כלומר מקדמים את i ב-2, וכעת $i = 5$. מתחילים לעשות השוואה נוספת בין התבנית מהתו האחרון שלה לטקסט, כשבעצם זו ההשוואה הראשונה שעשינו (חזרנו להתחלה).

שאלה 5. תיקון ראשון: האלגוריתם יעבוד, אך יעבוד אך יהיה פחות מהיר.

תיקון שני: האלגוריתם לא יעבוד. (דוגמא, כפי שראיתם בהרצאה: LAN–ILAN, טקסט: WILAN–ILAN).