פתרון ממ"ן 12

שאלה 1

א.

טיוטה

נוסחת הנסיגה המובאת בתרגיל נראית מתאימה למבנה המתואר במשפט האב. הדבר הראשון לעשות הוא לבדוק לאיזה מן המקרים של המשפט מתאימה הנוסחה.

בסימוני משפט האב, נקבל כי $a = 5, b = 2$ וכי $f(n) = n^2 \log n$ לכל $n \in \mathbb{N}$. על כן $\log_b a = \log_2 5 > 2.3$.

כעת עלינו לבדוק מהו ה"סדר" (האסימפטוטי) בין $f(n)$ ל $n^{\log_2 5}$.

לו הוכחנו כי עבור n גדול דיו $\log n \leq n^{0.2}$, למשל, אזי יכולנו להסיק כי $n^2 \log n = O(n^{2+0.2}) = O(n^{\log_2 5 - 0.1})$. יש לזכור כי באופן פורמלי מה שעלינו להוכיח הוא שקיים $n_0 \in \mathbb{N}$ וקיים $c > 0$ כך שלכל $n \geq n_0$ מתקיים $n^2 \log n \leq c \cdot n^{2.2}$. אם נסדר את האי שוויון נקבל כי עלינו להוכיח כי $\log n \leq cn^{0.2}$.

כמובן שעלינו להוכיח כל צעד כזה באופן פורמלי ומדויק.

תשובה

הטענה המרכזית שאנו מעוניינים להוכיח בסעיף זה היא הטענה הבאה.

משפט 1 $T(n) = \Theta(n^{\log_2 5})$.

לצורך כך נשתמש בטענה הראשונה שבמשפט האב.

נסמן $a = 5, b = 2$ וכמו כן נסמן $f(n) = n^2 \log n$ לכל $n \in \mathbb{N}$.

נתבונן ב $\varepsilon = 0.1$. משפט 1 נובע ממשפט האב ומהטענה הבאה.

טענה 2 $f(n) = O(n^{\log_2 5 - \varepsilon})$.

הוכחה על פי משפט ממדריך הלמידה (בעמוד 34) נקבל כי קיים $n_0 \in \mathbb{N}$ וקיים $c > 0$ כך שלכל $n \geq n_0$ מתקיים $\log n \leq cn^{0.2}$. נתבונן ב n_0 וב $c' = 2c > 0$. יהי $n \geq n_0$. אזי מכיוון ש $\log_2 5 > 2.3$,

$$f(n) = n^2 \log n \leq cn^{2.2} < cn^{\log_2 5 - \varepsilon} .$$

ומכאן כי $f(n) = O(n^{\log_2 5 - \varepsilon})$.

□

✓

ב.

טיוטה

גם נוסחת הנסיגה המובאת בתרגיל נראית מתאימה למבנה המתואר במשפט האב. שוב, הדבר הראשון לעשות הוא לבדוק לאיזה מן המקרים של המשפט מתאימה הנוסחה.

בסימוני משפט האב, נקבל כי $a = 7, b = 3$ וכי $f(n) = n^2$ לכל $n \in \mathbb{N}$. על כן $\log_b a = \log_3 7 < 1.8$.

כעת עלינו לבדוק מהו ה"סדר" (האסימפטוטי) בין $f(n)$ ל $n^{\log_3 7}$. מכיוון שלכל n מתקיים $f(n) = n^2 > n^{\log_3 7+0.1}$ נקבל כי אכן $f(n) = \Omega(n^{\log_3 7+0.1})$. ועל כן נרצה להשתמש בטענה השלישית שבמשפט האב. בל נשכח, אולם, כי במקרה זה עלינו להוכיח טענה נוספת. עלינו להוכיח כי קיימים $n_0 \in \mathbb{N}$ ו $0 < c < 1$ כך שלכל $n \geq n_0$ מתקיים $af(n/b) \leq cf(n)$.

תשובה

הטענה המרכזית שאנו מעוניינים להוכיח בסעיף זה היא הטענה הבאה.

משפט 3 $T(n) = \Theta(n^2)$.

לצורך כך נשתמש בטענה השלישית שבמשפט האב.

נסמן $a = 7, b = 3$ וכמו כן נסמן $f(n) = n^2$ לכל $n \in \mathbb{N}$. נתבונן ב $\varepsilon = 0.1$.

טענה 4 $f(n) = \Omega(n^{\log_3 7+\varepsilon})$.

הוכחה נתבונן ב $n_0 = 1$ וב $c = 1$. יהי $n \geq n_0$, אזי

$$f(n) = n^2 \geq n^{\log_3 7+\varepsilon}$$

$\square$

טענה 5 קיימים $0 < c < 1$ ו $n_0 \in \mathbb{N}$ כך שלכל $n \geq n_0$ מתקיים $af(n/b) \leq cf(n)$.

הוכחה נתבונן כעת ב $n_0 = 1$ וב $0 < c = \frac{7}{9} < 1$. לכל $n \geq n_0$ מתקיים $af(n/b) = 7\left(\frac{n}{3}\right)^2 \leq cf(n)$. $\square$

כעת נוכל להוכיח את משפט 3.

הוכחת משפט 3 על פי הטענה השלישית במשפט האב בצירוף טענות 4 ו 5 נקבל כי

$$T(n) = \Theta(f(n)) = \Theta(n^2) \ .$$

$\square$

$\checkmark$

ג.

טיוטה

גם נוסחת הנסיגה המובאת בתרגיל נראית מתאימה למבנה המתואר במשפט האב. שוב, הדבר הראשון לעשות הוא לבדוק לאיזה מן המקרים של המשפט מתאימה הנוסחה.

בסימוני משפט האב, נקבל כי $a = b = 2$ וכי $f(n) = n\log^2 n$ לכל $n \in \mathbb{N}$. על כן $\log_b a = \log_2 2 = 1$.

כעת עלינו לבדוק מהו ה"סדר" (האסימפטוטי) בין $f(n)$ ל n. בדומה לדוגמאות שראיתם בספר ובמדריך הלמידה, אכן מתקיים כי $f(n) \geq n$, אולם לא קיים $\varepsilon > 0$ כך שמתקיים $f(n) = \Omega(n^{1+\varepsilon})$. על כן לא ניתן לקבוע את סדר הגודל האסימפטוטי של T בעזרת משפט האב.

לפי שיטת האב המורחבת (עמוד 44 במדריך), עם זאת, ניתן להוכיח כי $T(n) = \Theta(n\log^3 n)$.

תשובה

הטענה המרכזית שאנו מעוניינים להוכיח בסעיף זה היא הטענה הבאה.

משפט 6 $T(n) = \Theta(n\log^3 n)$.

הוכחה על פי עקרון הרפלקסיביות $f(n) = n\log^2 n = \Theta(n^{\log_b a}\log^2 n)$. לפי שיטת האב המורחבת נובע כי $T(n) = \Theta(n\log^3 n)$.

□

✓

ד.

תשובה

נשתמש תחילה בנוסחת הנסיגה על מנת למצוא הצגה מפורשת של T. את הטענה הבאה ניתן להוכיח באינדוקציה על $n \in \mathbb{N}$ (ודאו זאת).

טענה 7 לכל $n \geq 2$ מתקיים $T(n) = T(1) + \sum_{j=2}^{n} j\log j$.

הטענה המרכזית שאנו מעוניינים להוכיח בסעיף זה היא הטענה הבאה.

משפט 8 $T(n) = \Theta(n^2\log n)$.

הוכחה נוכיח תחילה כי $T(n) = O(n^2\log n)$.

נתבונן ב $n_0 = 2$ וב $c = \max\{2, 2T(1)\} > 0$. יהי $n \geq n_0$, אז מכיוון ש $j\log j \leq n\log n$ לכל

$2 \le j \le n$, נקבל כי

$$T(n) = T(1) + \sum_{j=2}^{n} j \log j \le \frac{c}{2} + \sum_{j=2}^{n} n \log n$$
$$\le \frac{c}{2} + \frac{c}{2} n^2 \log n = \frac{c}{2}(1 + n^2 \log n) \le cn^2 \log n$$

מכאן כי $T(n) = O(n^2 \log n)$.

נוכיח כעת כי $T(n) = \Omega(n^2 \log n)$.

נתבונן ב $n_0 = 4$ וב $c = 1/8 > 0$. יהי $n \ge n_0$, אז מכיוון ש $j \log j \ge \frac{n}{2} \log \frac{n}{2}$ לכל $\frac{n}{2} \le j \le n$, ומכיוון ש $\log n \ge 2$ נקבל כי

$$T(n) = T(1) + \sum_{j=2}^{n} j \log j \ge \sum_{j=n/2}^{n} j \log j \ge \sum_{j=n/2}^{n} \frac{n}{2} \log \frac{n}{2}$$
$$= \left(\frac{n}{2}\right)^2 (\log n - 1) = \frac{n^2}{4} \log n - \frac{n^2}{4} \ge \frac{n^2}{4} \log n - \frac{n^2}{8} \log n = c \cdot n^2 \log n$$

מכאן כי $T(n) = \Omega(n^2 \log n)$. $\square$

$\checkmark$

ה.

<u>טיוטה</u>

התרגילים שבמדריך הלמידה מרגילים אותנו, שכאשר אנו רואים רקורסיה ובה $\sqrt{n}$, עלינו לחפש החלפת משתנים. גם הסעיף שלפנינו "תפור" לשיטה הזו. אכן, אם נסמן $S(m) = T(2^m)$ לכל $m \in \mathbb{N}$, נקבל כי $S(m) = S(m/2) + 5$.

ממשפט האב (טענה שנייה) נוכל לקבל כי $S(m) = \Theta(\log m)$ ועל כן אם נחליף משתנים שוב נקבל $T(n) = \Theta(\log \log n)$.

תשובה

הטענה המרכזית שאנו מעוניינים להוכיח בסעיף זה היא הטענה הבאה.

משפט 9 $T(n) = \Theta(\log \log n)$.

הוכחה נסמן $S(m) = T(2^m)$ לכל $m \in \mathbb{N}$, ונקבל את נוסחת הנסיגה$S(m) = S(m/2) + 5$. בסימוני משפט האב נסמן $a = 1, b = 2$, ו $f(m) = 5$ לכל m. מכיוון ש $f(n) = 5 = \Theta(1) = \Theta(n^{\log_2 1})$ נקבל ממשפט האב (מקרה שני) כי $S(m) = \Theta(\log m)$, ומכאן כי $T(n) = \Theta(\log \log n)$. $\square$

$\checkmark$

<u>**שאלה 2**</u>

טיוטה

כדאי לשים לב תחילה שבמקרה הפרטי בו $k = 2$ כבר נתקלנו באלגוריתם "מיון מיזוג". אלגוריתם המיזוג החזיק שני מצביעים, אחד לכל מערך. בכל איטרציה המצביעים הצביעו לאיבר הראשון (המינימלי) בכל מערך שטרם טיפלנו בו. האלגוריתם השווה בין האיברים המינימלים ובחר בקטן מביניהם.

הכללה של אותו האלגוריתם ל k כלשהו תניב אלגוריתם שזמן הריצה שלו $\Theta(nk^2)$. על מנת לשפר את זמן הריצה דרושה גישה עדינה יותר למציאת האיבר הקטן ביותר מבין אברי המינימום של כל המערכים.

מכיוון שבכל שלב אנחנו משנים את קבוצת אברי המינימום באיבר אחד (הקטן ביותר), הרעיון הוא להחזיק ערימת מינימום של האיברים המינימלים ובכל שלב להוציא את המינימום מן הערימה ולהכניס איבר חדש.

תשובה

האלגוריתם שנציג בתשובה זו פועל באופן הבא. תחילה האלגוריתם בונה ערימת מינימום מ k האיברים הראשונים במערכים הנתונים, ומקדם את המצביעים בכל המערכים להצביע על האיבר השני בכל מערך. כל איבר בערימה, $heapElement$ נוסף על המפתח שלו $key[heapElement]$ מחזיק שדה נוסף $arr[heapElement]$ שהוא מספר המערך ממנו הגיע האיבר. האלגוריתם מאתחל מערך חדש A באורך nk ומחזיק מצביע $index$ לתחילת המערך. בכל שלב במהלך הריצה, האלגוריתם שולף את האיבר הקטן ביותר מתוך הערימה, ומכניס אותו למערך A במקום $index$. האלגוריתם בודק לאיזה מ k המערכים שייך האיבר הקטן ביותר, ומכניס את האיבר הבא במערך לערימה. האלגוריתם עוצר כאשר הערימה ריקה. נציג כעת את האלגוריתם באופן פורמלי.

```
mergeManyToOne(A_1, ..., A_k)

create an empty array A of size nk. let index ← 1
create an empty minimum-heap H of size k.
for j = 1 to k
        let index_j ← 1
        min-heap-insert(H, heapElement(A_j[index_j], j))
        index_j ← index_j + 1
while H is not empty
           let tempElement ←heap-extract-min(H).
           A[index] ← key[tempElement].
           index ← index + 1
           let j ← arr[tempElement].
           min-heap-insert(H, heapElement(A_j[index_j], j))
           index_j ← index_j + 1
return A.
```

הטענה המרכזית שאנחנו רוצים להוכיח על האלגוריתם היא הטענה הבאה

משפט 10 (נכונות האלגוריתם) בהנתן מערכים ממויינים $A_1, \ldots, A_k$, האלגוריתם מחזיר מערך ממויין A, שהוא מיזוג של המערכים הנתונים.

הוכחה נסמן ב A' את המערך באורך nk המכיל את אברי כל המערכים לפי סדר., ונוכיח כי בסיום הריצה, $A[j] = A'[j]$ לכל $1 \le j \le nk$.

לשם כך נוכיח כעת באינדוקציה על $1 \le i \le nk$ כי בסיום האיטרציה ה i של האלגוריתם מתקיים כי (1) $A[s] = A'[s]$ לכל $1 \le s \le i$; וגם (2) $A'[i+1]$ נמצא ב H.

נוכיח תחילה את הטענה עבור $i = 1$. בתחילת הריצה, האלגוריתם מכניס את כל האיברים הראשונים במערכים לערימה. אחד מהאיברים האלה הוא האיבר הקטן ביותר מבין אברי כל המערכים. באיטרציה הראשונה מוצא האלגוריתם את האיבר המינימלי מן הערימה, ומכניס אותו ל $A[1]$. מכיוון שזהו האיבר הקטן ביותר, נובע כי $A[1] = A'[1]$. נסמן ב j את האינדקס של המערך ממנו הגיע $A[1]$. כעת מכניס האלגוריתם לערימה את $A_j[2]$. נניח בשלילה כי $A'[2]$ איננו בערימה. אז קיים j_0 וקיים r_0 כך ש

$A'[2] = A_{j_0}[r_0]$. אם $j_0 = j$, אז מההנחה בשלילה $r_0 > 2$ ונקבל כי $A_j[r_0] > A_j[2]$ בסתירה לכך שהמערך ממויין. בדומה אם $j_0 \neq j$. על כן $A'[2]$ נמצא בערימה.

נניח נכונת הטענה עבור i ונוכיח עבור $i+1$. לפי הנחת האינדוקציה נובע כי $A'[i+1]$ נמצא בערימה וכי $A[s] = A'[s]$ לכל $1 \leq s \leq i$. על כן $A'[i+1]$ הוא האיבר המינימלי בערימה, ועל כן $A[i+1] = A'[i+1]$. ההוכחה כי $A'[i+2]$ נמצא בערימה בסוף האיטרציה דומה למקרה $i = 1$.

מהטענה שהוכחנו נובע כי בסיום ההרצה ה nk מתקיים $A[j] = A'[j]$ לכל $1 \leq j \leq nk$. □

הטענה השנייה שנרצה להוכיח היא הטענה הבאה.

משפט 11 (סיבוכיות האלגוריתם) זמן הריצה של האלגוריתם במקרה הגרוע ביותר הוא $\Theta(nk \log k)$.

הוכחה שתי השורות הראשונות מתבצעות בזמן קבוע. שורות 4 ו 6 מתבצעות בזמן קבוע וזמן הריצה של שורה 5 הוא לכל היותר $O(\log k)$. לכן הלולאה בשורות 3-6 מתבצעת בזמן לכל היותר $O(k \log k)$. שורות $9, 10, 11, 13$ מתבצעות בזמן קבוע ושורות $8, 12$ מתבצעות בזמן $O(\log k)$, ועל כן הלולאה בשורות $7 - 13$ מתבצעת בזמן $O(nk \log k)$ (nk איטרציות בזמן ריצה של $O(\log k)$ כל אחת).

מכאן כי זמן הריצה של האלגוריתם הוא $O(nk \log k)$.

על הקלט הבא האלגוריתם יבצע $\Omega(nk \log k)$ פעולות (ודאו זאת!).

$$A_1 = [k, 2k, 3k, ..., nk]$$
$$A_2 = [k-1, 2k-1, ..., nk-1]$$
$$\vdots$$
$$A_k = [1, 2, ..., k]$$

לכן זמן הריצה של האלגוריתם הוא $\Theta(nk \log k)$. □

✓

<u>שאלה 3</u>

תשובה

האלגוריתם שנציג מקבל מערך A באורך n ומספר $k < \frac{n}{\log n}$ ובונה מ A ערימת מינימום H. בשלב הבא מוציא האלגוריתם k פעמים את האיבר המינימלי מ H. נציג את האלגוריתם באופן פורמלי.

kSmallest(A, k)

1. create a minimum-heap H from A.
2. let $temp \leftarrow 0$
3. **for** $j = 1$ to k
4. $\quad temp \leftarrow$heap-extract-min(H)
5. **return** $temp$.

הטענה המרכזית שאנחנו רוצים להוכיח על האלגוריתם היא הטענה הבאה

משפט 12 (נכונות האלגוריתם) בהנתן מערך A ומספר k, האלגוריתם מחזיר את האיבר ה k קטן ביותר ב A.

לשם הוכחת משפט 12 נוכיח באינדוקציה כי בסיום האיטרציה ה j, עבור $1 \leq j \leq k$, $temp$ שווה לאיבר ה j קטן ביותר ב A (ודאו זאת!).

הטענה השנייה שנרצה להוכיח היא הטענה הבאה.

משפט 13 (סיבוכיות האלגוריתם) זמן הריצה של האלגוריתם במקרה הגרוע ביותר הוא $\Theta(n)$.

הוכחה השורה הראשונה מתבצעת בזמן $O(n)$. שורה 4 מתבצעת בזמן $O(\log n)$ ועל כן זמן הריצה של האלגוריתם הוא לכל היותר $O(n + k\log n)$. מתוך ההנחה כי $k < \frac{n}{\log n}$ נובע כי זמן הריצה הוא $O(\log n)$.

על הקלט הבא האלגוריתם build-min-heap מבצע $\Omega(n)$ פעולות(ודאו זאת!), ועל כן האלגוריתם שלנו יבצע $\Omega(n)$ פעולות (כבר מן השורה הראשונה).

$$A = [1, 2, \ldots, n/4, n/2+1, n/2+1, ..., n, n/4+1, ..., n/2]$$

מכאן כי זמן הריצה של האלגוריתם הוא $\Theta(n)$.

$\square$

$\checkmark$