מבני נתונים
פתרון תרגיל 8

גלעד אשרוב

11 ביוני 2013

תרגיל 0. (תרגיל זה הוא בשבילכם, ולא להגשה). הכנס את האיברים הבאים לעץ AVL:
$20, 12, 15, 17, 23, 53, 11, 10, 32, 45, 44, 22, 2, 9, 4$.
לאחר מכן, ביחרו שני איברים כלשהם והוציאו אותם מהעץ.

תרגיל 1. משפחה של עצים תיקרא "מאוזנת" אם כל עץ במשפחה הוא בעל גובה $O(\log n)$ רמות, כאשר n הוא מספר הקודקודים בעץ. נגדיר את הפונקציה $f : V \to \mathbb{N}$ בצורה הבאה:

$$f(u) = \begin{cases} n/2^i & \text{if } i < \log n \\ 1 & \text{otherwise} \end{cases},$$

כאשר i הוא העומק של הקודקוד u (עומקו של השורש הוא 0), ונניח ש-n הוא חזקה שלמה של 2.

1. נתונה משפחה של עצים בינאריים שבה לכל עץ T מתקיים: $\sum_{u \in T} f(u) \leq n \log n$. האם המשפחה מאוזנת? הוכח/י את תשובתך.

2. כיצד תשובתך הייתה משתנה אם במקום הערך 1 בתנאי האחרת, היה הערך 2? הוכח/י את תשובתך.

פתרון. השאלה יצאה קלה ממה שהתכוונתי (קורה...). התכוונתי לכך שמתקיים כי: $\sum_{u \in T} f(u) \in \Theta(\log n)$. פתרו את השאלה עם השינוי הנ"ל לפני המבחן.
לגבי השאלה שפורסמה, קל להפריך:
נשים לב שהתכונה היא אסימפטוטית. כלומר, צריך להראות שעבור כל n_0 שבוחרים, ניתן למצוא דוגמא נגדית עבור $n \geq n_0$. אצלינו ־ נראה פשוט דוגמא נגדית לכל n. לכל n נקח פשוט עץ שרוך מאורך n. יתקיים כי עד ל-$\log n$ קודקודים הערכים $f(u)$ יהיו גדולים מ-1, והחל מ-$\log n$ כולם יהיו 1. נקבל אם כן:

$$\sum_{u \in T} f(u) = (n + n/2 + n/4 + \ldots + 1) + 1 \cdot (n - \log n) \leq 2n + (n - \log n) = 3n - \log n \leq 3n \leq n \log n$$

החל מ-$n_0 = 8$. מצד שני ־ ברור שהעץ לא מאוזן.
לגבי סעיף ב' - אותו הדבר:

$$\sum_{u \in T} f(u) = (n + n/2 + n/4 + \ldots + 1) + 2 \cdot (n - \log n) \leq 2n + 2(n - \log n) \leq 4n \leq n \log n$$

החל מ-$n_0 = 16$.

תרגיל 2. נגדיר כגלגול את הפעולה שנעשית באיזון מחדש של עצי AVL. ראינו 4 סוגי גלגולים: פשוט וכפול, ימינה ושמאלה.

א. הוכח/י שבהינתן 2 עצי חיפוש בינאריים T_1 ו-T_2 בעלי אותם n ערכים ניתן להגיע מ-T_1 ל-T_2 ע"י ביצוע סדרה של גלגולים פשוטים.

רמז: הראה/י קודם איך לטפל בשורש ואח"כ בתת העצים.

ב. הראה/י שמספר הגלגולים הדרוש הוא $O(n^2)$.

פתרון: נתבונן בשורש של-T_2. נניח שהשורש שלו הוא r. נחפש את השורש r בעץ T_1, ונבצע סידרה של גלגולים פשוטים בכדי להעלות אותו לשורש (קל..). כעת, מכיוון ששניהם העצים הם עצי חיפוש עם אותם הערכים - גודל תת העץ הימני של T_1 מכיל את כל המפתחות שקטנים מ-r, בדיוק כמו ב-T_2. בצורה דומה - עבור תת העץ הימני של T_1 ותת העץ הימני של T_2. נריץ את האלגוריתם רקורסיבית על שני תתי העצים $(L(T_1), L(T_2))$, ובנוסף $(R(T_1), R(T_2))$. הנ"ל בעצם יחזיר לנו את הגלגולים שצריך לעשות בכדי להעביר את $L(T_1)$ ל-$L(T_2)$, וכן את הגלגולים שצריך לעשות בכדי להעביר את $R(T_1)$ ל-$R(T_2)$. בסופו של דבר, נחזיר את סדרת הגלגולים הנדרשים כדי להעביר את השורש של T_1 להיות אותו השורש כמו T_2 + שתי סדרות הגלגולים המתאימים לתתי העצים.

האלגוריתם הנ"ל מראה שאפשר לעבור מ-T_1 ל- T_2, ואפילו מראה כיצד ניתן למצוא את סידרת הגלגולים המתאימה.

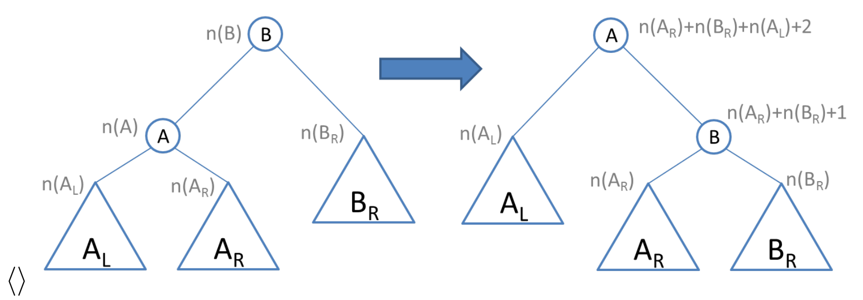
לגבי הסעיף השני, במקרה הגרוע ביותר העצים ממש לא מאוזנים ונצטרך לבצע סדרה של n גלגולים בכדי להעלות את השורש. מקרה זה מתקבל כאשר השורש הוא האיבר הגדול ביותר / הקטן ביותר. אם כך, הנוסחא הרקורסיבית תהיה:

$$T(n) = T(1) + T(n-1) + n$$

ופתרון הנוסחא הנ"ל הוא $O(n^2)$.

תרגיל 3. נניח שבעץ AVL שומרים לכל צומת בנוסף לשדות הרגילים, את הדרגה של הצומת (מספר הקודקודים בתת העץ המושרש באותו הצומת). הראו כיצד ניתן לתחזק את השדות הנ"ל עם הכנסות והוצאות ב-AVL. כלומר, הראו כיצד ניתן לשנות את השדות הנ"ל עם כל הכנסה והוצאה.

פתרון: בזמן הכנסה נגדיל את $n(v)$ ב-1 לאורך המסלול מהשורש לעלה שהוכנס. בזמן הסרה - נוריד באחד. בזמן גלגול - נעדכן את השדה כנדרש. למשל, בגלגול LL:



תרגיל 4. תזכורת: האינטרוול $I = [x, y]$ (עבור $x \le y$) היא קבוצת כל המספרים שבין x ל-y, כלומר -

$$\{z \mid x \le z \le y\} \; .$$

לערך x אנחנו קוראים $I.low$ ולערך y נקרא $I.high$.

נניח ואנו רוצים לבנות מבנה נתונים המאחסן אינטרוולים. בנוסף לפעולות הכנסה והוצאה, נרצה לתמוך בשאילתא הבאה:

חיפוש אינטרוול $[a, b]$: האם קיים במבנה אינטרוול $[x, y]$ שמכיל את $[a, b]$. כלומר, אינטרוול $[x, y]$ המקיים: $x \leq a \leq b \leq y$. אם כן - החזר אותו.

בכדי לעשות זאת, נשמור את הנתונים בעץ AVL לפי מפתח - low. בנוסף, לכל קודקוד x בעץ, נשמור משתנה נוסף M שיישמור את המקסימום של ערכי ה-$high$ של כל תת העץ המושרש בקודקוד x, כלומר, המשתנה ישמור את:

$$x.\mathsf{M} = \max\{x.\mathsf{interval.high},\ x.\mathsf{left.M},\ x.\mathsf{right.M}\}\ .$$

הראו כיצד לממש את הפעולה "חיפוש אינטרוול $[a, b]$", וכיצד השדה M עוזר לנו בחיפוש זה.

פתרון: חיפוש ייעשה בצורה הבאה:

Algorithm 1 $Interval - Search(i)$

```
x ← root.
while x ≠ NULL and (i.low > x.interval.high or x.interval.low > i.high) do
   if x.left ≠ NULL and i.low ≤ left.x.M then
      x =x.left
   else
      x = x.right
   end if
end while
```