

מבני נתונים

תרגיל 4 - פתרון

גלעד אשרוב

2 ביוני 2014

תרגיל 1. לסעיפים הבאים, כתבו אלגוריתמים הכי יעילים (אסימפטוטית) למשימה, והסבירו מדוע לא ניתן לבנות אלגוריתם יעיל יותר:

1. כתבו אלגוריתם המקבל כקלט עץ בינארי, ומחזיר האם הוא עץ חיפוש.
2. כתבו אלגוריתם המקבל כקלט עץ בינארי, ומחזיר האם הוא עץ AVL .

פתרון:

1. ניזכר שעץ חיפוש הוא עץ המכיל שני תתי עצים שהם בעצמם עצי חיפוש, וכל המפתחות בתת העץ השמאלי קטנים מערכו של המפתח בשורש, שקטן מכל המפתחות בתת העץ השמאלי. בכדי לדעת האם העץ הוא עץ חיפוש - נשתמש בפונקציה רקורסיבית המקבלת עץ ומחזירה שלושה ערכים - האם העץ הוא עץ חיפוש, מהו ערך המפתח הגדול ביותר בעץ, ומהו ערך המפתח הקטן ביותר.
האלגוריתם:

- (א) אם העץ הוא עלה בודד בעל הערך x , החזר (yes, x, x) .
- (ב) אחרת - נניח העץ הוא (T_{left}, r, T_{right}) , כאשר T_{left} הוא תת העץ השמאלי, r הוא ערכו של השורש, ו- T_{right} הוא תת העץ הימני. בצע:
- i. הפעל רקורסיבית את האלגוריתם על T_{left} , וקבל חזרה $(b_{left}, \min_{left}, \max_{left})$.
(אם T_{left} ריק, נאמר שקיבלנו חזרה (yes, r, r) .)
 - ii. הפעל רקורסיבית את האלגוריתם על T_{right} , וקבל חזרה $(b_{right}, \min_{right}, \max_{right})$.
(אם T_{right} ריק, נאמר שקיבלנו חזרה (yes, r, r) .)
 - iii. אם $b_{left} = b_{right} = true$, וגם $\max_{left} \leq r \leq \min_{right}$, החזר: $(true, \min_{left}, \max_{right})$. אחרת, החזר $(false, 0, 0)$.

קל לראות שעלות האלגוריתם הוא $O(n)$.

2. ראשית, נבדוק שהעץ הוא באמת עץ חיפוש, ולשם כך נפעיל את האלגוריתם מהסעיף הקודם. כעת, נרצה שהוא יהיה גם עץ AVL . לשם כך, נכתוב פונקציה רקורסיבית המקבלת עץ ומחזירה שני ערכים - האם העץ הוא AVL , ואת גובה העץ. נקבל אם כן:

- (א) אם העץ הוא עץ ריק, החזר $(yes, -1)$.
- (ב) אחרת, נניח העץ הוא (T_{left}, r, T_{right}) , כאשר T_{left} הוא תת העץ השמאלי, ו- T_{right} הוא תת העץ הימני. אזי:

- i. הפעל רקורסיבית את האלגוריתם על תת העץ השמאלי. קבל חזרה $(b_{\text{left}}, h_{\text{left}})$.
- ii. הפעל רקורסיבית את האלגוריתם על תת העץ השמאלי. קבל חזרה $(b_{\text{right}}, h_{\text{right}})$.
- iii. אם $b_{\text{left}} = b_{\text{right}} = \text{true}$, וגם $|h_{\text{left}} - h_{\text{right}}| \leq 1$, אזי החזר $(\text{true}, \max\{h_{\text{left}}, h_{\text{right}}\} + 1)$. אחרת, החזר $(\text{false}, 0)$.

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

א. הוכחי/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 סדרת הגלגולים המתאימה.

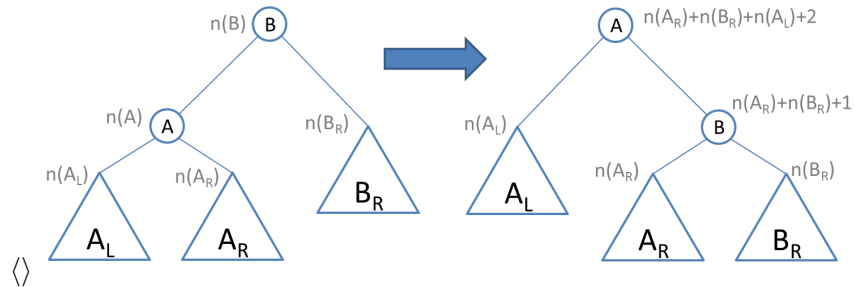
לגבי הסעיף השני, במקרה הגרוע ביותר העצים ממש לא מאוזנים ונצטרך לבצע סדרה של n גלגולים בכדי להעלות את השורש. מקרה זה מתקבל כאשר השורש הוא האיבר הגדול ביותר / הקטן ביותר. אם כך, הנוסחה הרקורסיבית תהיה:

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

ופתרון הנוסחה 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 $x \leq y$) היא קבוצת כל המספרים שבין x ל- y , כלומר -

$$\{z \mid x \leq z \leq y\} .$$

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 בשאלתא 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:

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

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)

```

 $x \leftarrow root.$ 
while  $x \neq NULL$  and ( $i.low > x.interval.high$  or  $x.interval.low > i.high$ ) do
  if  $x.left \neq NULL$  and  $i.low \leq left.x.M$  then
     $x = x.left$ 
  else
     $x = x.right$ 
  end if
end while

```

תרגיל 5. עץ חיפוש בינארי ייקרא "עץ אדום שחור" אם הוא עץ שלם¹ ומקיים את התכונות הבאות:

1. כל צומת צבוע ב"אדום" או "שחור".

2. כל עלה צבוע ב"שחור".

¹**תזכורת:** עץ שלם הוא עץ שבו כל צומת הוא או עלה, או שיש לו שני בנים.

3. אם צומת הוא אדום, אז שני בניו שחורים.

4. כל המסלולים מכל קודקוד לכל העלים שהם צאצאיו, מכילים את אותו המספר של צמתים שחורים.

ענה על הסעיפים הבאים:

א. מה ניתן להגיד על צבעם של הקדקדים במסלול מהשורש לעלה כלשהו? (מהו יחס כמות הצמתים השחורים לעומת האדומים?)

הבהרה: עליכם לחשוב על עקרה כללי ולא להתמקד בהכרח בדוגמא שלפניכם.

ב. נגדיר את "הגובה השחור" כדלהלן:

הגדרה 1. (גובה שחור) לכל צומת x נגדיר "גובה השחור" ($black\ height$) של x , כמספר הקודקודים השחורים בכל מסלול מצומת x ועד לעלה (לא כולל את הצומת x עצמו). נסמן מספר זה ב- $bh(x)$.

בדוגמא שלפניכם, עבור הקודקוד 8, גובהו השחור הוא: $bh(8) = 2$. בצורה דומה, ציין לכל קודקוד את הגובה השחור שלו.

ג. נניח שבעץ h רמות, ויהי x השורש. הסבר מדוע: $bh(x) \geq h/2$.

ד. יהי x קודקוד פנימי, ויהיו y, z שני בניו. הוכח כי: $bh(x) \leq bh(y) + 1$ (ובאותו אופן גם: $bh(x) \leq bh(z) + 1$).

ה. הוכח את הטענה הבאה: לכל קודקוד x בעץ אדום שחור, תת העץ המושרש ב- x (כלומר, תת העץ ש- x הוא השורש שלו) מכיל לפחות $2^{bh(x)} - 1$ צמתים פנימיים. (ההוכחה היא באינדוקציה על גובהו של x , והשתמשו בסעיף הקודם).

ו. השתמש בסעיפים ג' ו-ה' בכדי להראות שגובהו של עץ אדום שחור בעל n צמתים פנימיים הוא לכל היותר $2 \log(n + 1)$.

פתרון:

א. נשים לב שאם צומת הוא אדום, אזי שני בניו שחורים. כמו כן, מספר הצמתים השחורים בכל המסלולים בין השורש לכל אחד מהעלים הוא שווה. לכן, במסלול הכי קצר מהשורש לעלה, כל הצמתים יהיו שחורים. במסלול הכי ארוך - יהיו מקסימום פי שניים צמתים - אותו מספר של צמתים שחורים כמו במסלול הכי קצר, ומקסימום בין כל שני שחורים יימצא קודקוד אחד אדום (שימו לב שאי אפשר שיהיו שני קודקודים אדומים ברצף).

ב. בדוגמא הנתונה לנו, נקבל כי:

• $bh(x) = 0$: קודקודים אלו הם כל העלים (NIL).

• $bh(x) = 1$: הקודקודים 1, 6, 11, 15, 22, 25, 27 (נשים לב שהגדרת $bh(x)$ אומרת כל הקודקודים השחורים עד לעלה - מלבד הקודקוד x עצמו), ולכן גם הצמתים 1, 11, 15, 25 הם בעלי $bh(x) = 1$ אף על פי שהם שחורים בעצמם.

• $bh(x) = 2$: 8, 17, 13.

ג. דומה ל-א': נניח שהגובה הוא h . לפי כלל (3) (לכל אדום - שני בניו שחורים) נקבל שבכל מסלול פשוט מהשורש לעלה יש לכל הפחות $h/2$ צמתים שחורים, ולכן $bh(x) \geq h/2$ (כאשר x מסמן את השורש).

ד. נתבונן על x ו- y . נשים לב כי:

- אם y אדום, אזי $bh(x) = bh(y)$. זאת מכיוון ש $bh(x)$ הוא מספר השחורים בעץ המורש ב- x , לא כולל הצומת x עצמו.
- אם y שחור, אזי $bh(x) = bh(y) + 1$. (שכן, y שחור, ואינו נספר ב- $bh(y)$, אבל כן נספר ב- $bh(x)$)

לכן, בסה"כ, לא משנה מהו צבעו של y , בכל אופן מתקיים כי: $bh(x) \leq bh(y) + 1$.

ה. הוכחה: ההוכחה היא באינדוקציה על גובהו של x . אם x הוא עלה, אזי $bh(x) = 0$, ותת העץ שלו אינו מכיל צמתים פנימיים. הטענה אומרת שתת העץ מכיל $2^0 - 1 = 0$ צמתים פנימיים, ולכן הטענה מתקיימת.

צעד האינדוקציה: נניח ש- x הוא בעל גובה חיובי. נתבונן בתת עץ ששורשו ב- x . מכיוון ש- x אינו עלה, יש לו שני בנים, נניח y, z . כפי שראינו, $bh(x) \leq bh(y) + 1$ וגם $bh(x) \leq bh(z) + 1$. לסיכום, נוכל לומר כי $bh(y) = bh(z) \geq bh(x) - 1$.

על כל פנים, גם y וגם z , שניהם בעלי רמה נמוכה יותר מהרמה של x , ולכן ניתן להפעיל עליהם את הנחת האינדוקציה. לפי ההנחה, $|T(y)| \geq 2^{bh(y)} - 1$. לפי המסקנה שראינו, נקבל כי $|T(x)| \geq 2^{bh(x)-1} - 1$, ובאופן דומה $|T(z)| \geq 2^{bh(x)-1} - 1$. נקבל אם כן:

$$|T(x)| = |T(y)| + |T(z)| + 1 \geq 2 \cdot (2^{bh(x)-1} - 1) + 1 = 2^{bh(x)} - 2 + 1 = 2^{bh(x)} - 1$$



וזה בדיוק מה שהיינו צריכים להוכיח.

ו. יהי x השורש, ויהי h הגובה. לפי מה שראינו, $bh(x) \geq h/2$ וגם $|T(x)| \geq 2^{bh(x)} - 1$. נסמן $|T(x)| = n$. נקבל אם כן:

$$\begin{aligned} n &\geq 2^{bh(x)} - 1 \\ n + 1 &\geq 2^{bh(x)} \geq 2^{h/2} \\ \frac{h}{2} &\leq \log(n + 1) \\ h &\leq 2 \log(n + 1) \end{aligned}$$