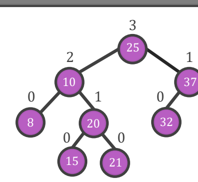
עצי AVL
AVL Trees

עץ AVL הגדרה

עץ AVL הוא עץ חיפוש בינארי שבו לכל צומת v התכונה:
הפרש הגבהים בין תת-עץ השמאלי לתת-העץ הימני של v הוא לכל היותר 1

דוגמא



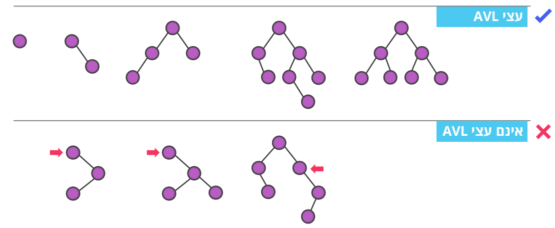
עץ AVL
מקור השם

- הנקרא על שם ממציאיו Adelson-Velskii ו-Landis
- היו הראשונים שהציעו עצים מאוזנים בצורה דינאמית ב-1962

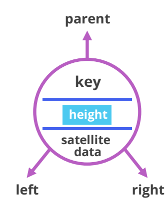
Adelson-Velskii

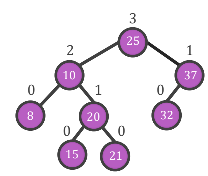
Landis

עץ AVL
דוגמאות



ייצוג
אפשרות ראשונה



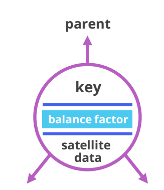


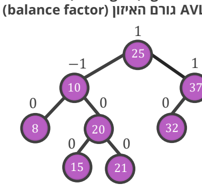
לייצג שדה גובה נדרשים $\log\log n$ ביטים

ייצוג
אפשרות שניה

balance factor = height (left subtree) – height (right subtree)

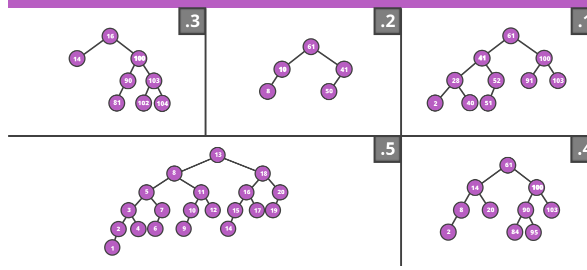
בעץ AVL גורם האיזון (balance factor) הוא ‎-1, 0 או 1





לייצג שדה balance factor נדרשים רק 2 ביטים

?אילו מהעצים הבאים הם עצי AVL



משפט

גובה של עץ AVL בעל n צמתים הינו $O(\log n)$

הוכחה:

- נעבוד על הבעיה "ההפוכה" – נמצא חסם תחתון על מספר המינימאלי של צמתים בעץ AVL בגובה h
- נסמן ב- n_h את המספר המינימאלי של צמתים בעץ AVL בעל גובה h
- נראה ש- n_h גדל בקצב מעריכי, כלומר נראה ש- $n_h = \Omega(c^h)$ עבור קבוע $c > 1$
- מכאן נסיק כי גובה של עץ AVL בעל n צמתים חסום על ידי $O(\log n)$

הוכחה המשך

עבור $n_0 = 1, h = 0$

עבור $n_1 = 2, h = 1$

עבור $n_2 = ?, h = 2$

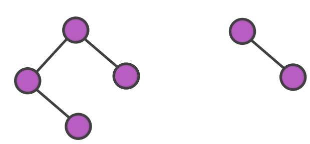
?

?n_2 למה שווה

1. $n_2 = 3$
2. $n_2 = 4$
3. $n_2 = 5$
4. $n_2 = 7$

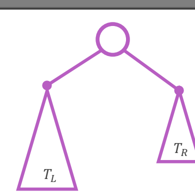
הוכחה המשך

עבור $n_0 = 1, h = 0$ | עבור $n_1 = 2, h = 1$ | עבור $n_2 = ?, h = 2$



הוכחה המשך

$h \geq 3$

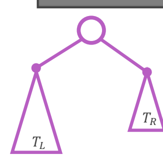


בעץ AVL בעל גובה h ומספר מינימאלי אפשרי של צמתים,
מהו הפרש הגבהים בין תת עץ השמאלי ותת עץ הימני של השורש?

1. 1
2. 0
3. יכול להיות כל מספר בין 0 ל $h-1$

הוכחה המשך

$h \geq 3$



הוכחה המשך

מסקנה מהמשפט

בעץ AVL, זמן ריצה של הפעולות

- *Search*
- *Min*
- *Max*
- *Successor*
- *Predecessor*

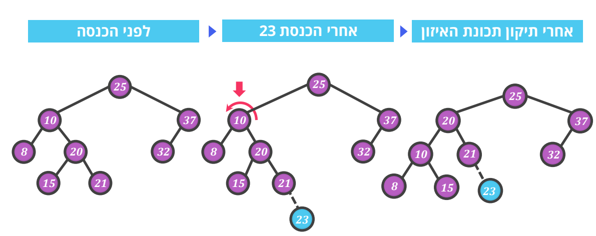
הינו $O(\log n)$ במקרה הגרוע (כאשר n הוא מספר הצמתים בעץ)

עץ AVL

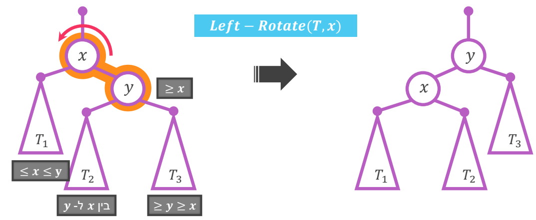
- גובה של עץ AVL בעל n צמתים הינו $O(\log n)$
- בעץ AVL, זמן ריצה של הפעולות
  - *Search*
  - *Min*
  - *Max*
  - *Successor*
  - *Predecessor*

  הוא $O(\log n)$ במקרה הגרוע

דוגמה

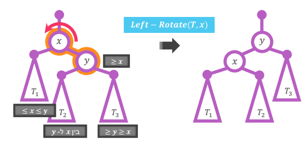


רוטציה שמאלית



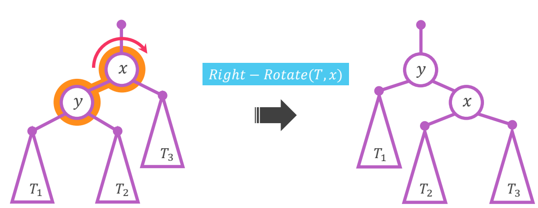
רוטציה שמאלית
פסאודו-קוד

```
Left − Rotate (T,x)
y ← x.right
x.right ← y.left
if y.left ≠ NULL
      (y.left).parent ← x
y.parent ← x.parent
if x.parent = NULL
      T.root ← y
else if x = (x.parent).left
      (x.parent).left ← y
else
      (x.parent).right ← y
y.left ← x
x.parent ← y
```



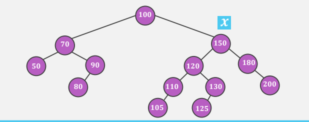
זמן ריצה $O(1)$

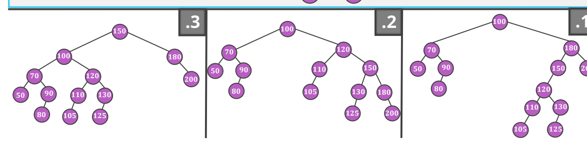
רוטציה ימנית



נתון עץ חיפוש בינארי T שלהלן.
איזה מבין העצים הוא עץ שמתקבל אחרי ביצוע פעולה $Left - Rotate(T,x)$
על צומת x בעל מפתח 150?

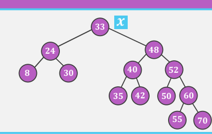
נתון: T

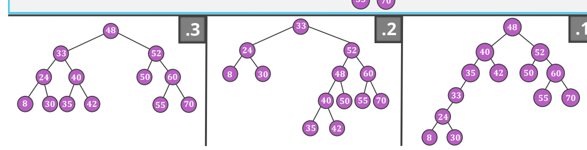




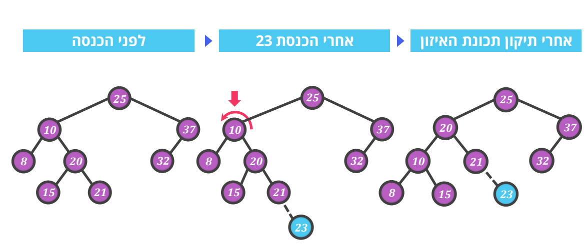
נתון עץ חיפוש בינארי T שלהלן.
איזה מבין העצים הוא עץ שמתקבל אחרי ביצוע פעולה $Left - Rotate(T,x)$
על שורש העץ?

נתון: T





הכנסה



נכון או לא נכון:

טענה: הצמתים היחידים שאולי הופרה בהם תכונת האיזון הם צמתים לאורך מסלול הכנסה.

1. הטענה נכונה
2. הטענה לא נכונה

נכון או לא נכון

טענה: אם עבור צומת v במסלול הכנסה, גובה של תת עץ המושרש ב- v לא השתנה, אז גורמי האיזון בצמתים שמעליו לא השתנו.

1. הטענה נכונה
2. הטענה לא נכונה

נכון או לא נכון

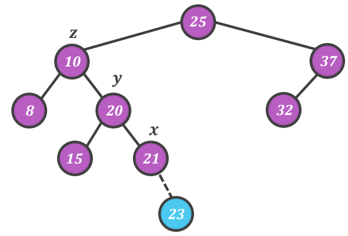
נניח שבעקבות ההכנסה, צומת v בעץ הפך להיות לא מאוזן.

טענה: גורם האיזון לא יכול להיות גדול מ- 2 בערכו המוחלט, כי בהכנסה גובה של הצומת יכול לגדול ב-1 לכל היותר.

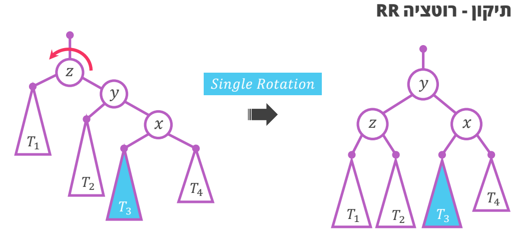
1. הטענה נכונה
2. הטענה לא נכונה

הכנסה

אחרי הכנסה 23



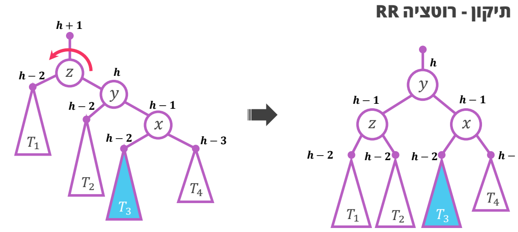
תיקון - רוטציה RR



y בן ימני של z

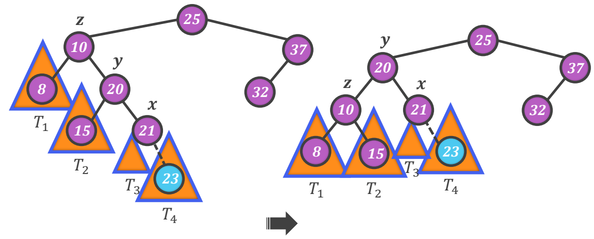
x בן ימני של y

תיקון - רוטציה RR

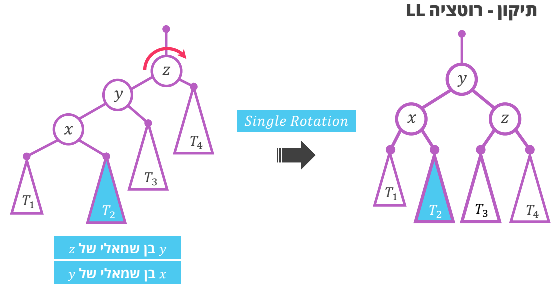


h – גובה של z לפני ההכנסה

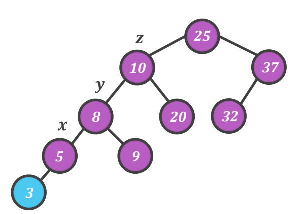
רוטציה RR
דוגמא



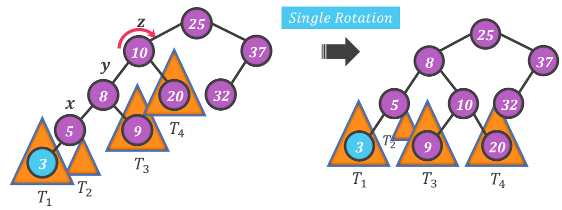
תיקון - רוטציה LL



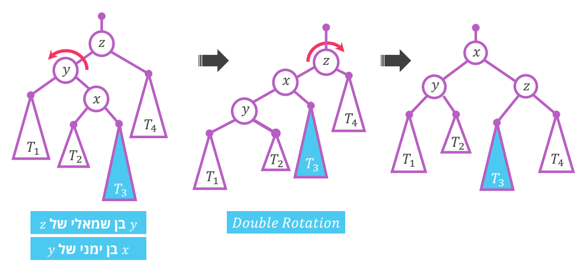
הכנסה
הכנסה 3



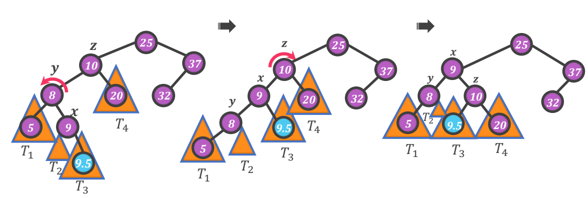
רוטציה LL
דוגמה



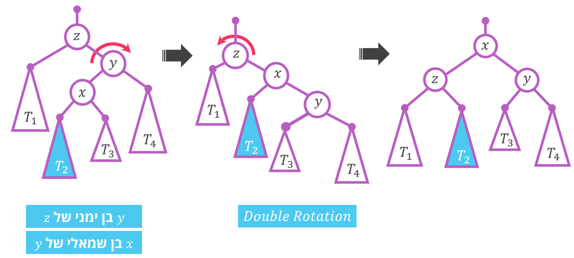
תיקון - רוטציה LR



רוטציה LR
דוגמה

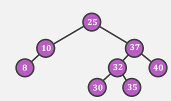


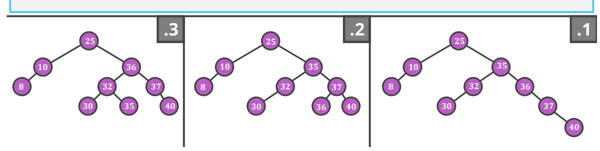
תיקון - רוטציה RL



מהו העץ שמתקבל לאחר הכנסת צומת בעל מפתח 36 לעץ AVL שבציור?

נתון: T





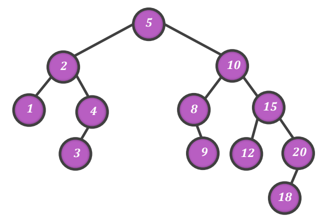
הכנסה בעץ AVL

- **הכנסת צומת לפי אלגוריתם ההכנסה של BST** $O(\log n)$
- **נטפס מהעלה שהוכנס במסלול לשורש, לכל צומת בדרך נעדכן את שדה הגובה (או balance factor) ונבדוק האם יש הפרה של תכונת ה-AVL** $O(\log n)$
  - הפרה של תכונת ה-AVL תתרחש אם גורם האיזון של הצומת הפך להיות 2 או -2
- **במידה וצומת v במסלול הכנסה אינו מקיים את תכונת האיזון, יש לבצע רוטציה מתאימה**
  - פעולת הרוטציה מחזירה את הגובה של תת-העץ שעל שורשו היא פועלת לגובה שהיה לפני ההכנסה, ולכן העץ הופך להיות מאוזן לאחר הרוטציה $O(1)$

זמן ריצה הכולל: $T(n) = O(\log n)$

מחיקה

מחיקה 1



טענה: הצמתים היחידים שאולי הופרה בהם תכונת האיזון הם צמתים לאורך מסלול מאבא של הצומת שנמחק בעלייה לשורש.

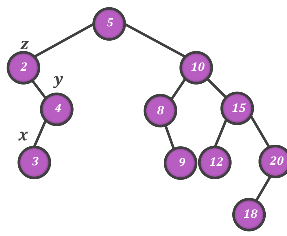
1. הטענה נכונה
2. הטענה לא נכונה

נניח שבעקבות המחיקה, צומת v בעץ הפך להיות לא מאוזן.
טענה: גורם האיזון לא יכול להיות גדול מ- 2 בערכו המוחלט, כי במחיקה גובה של הצומת יכול לקטון ב-1 לכל היותר.

1. הטענה נכונה
2. הטענה לא נכונה

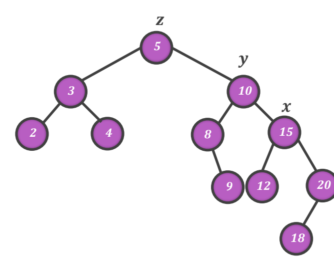
מחיקה

רוטציה RL

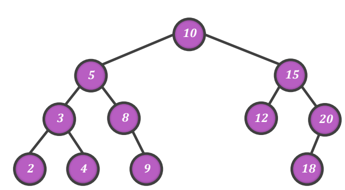


מחיקה

רוטציה RR



מחיקה



מחיקת 1

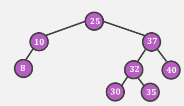
מחיקה בעץ AVL

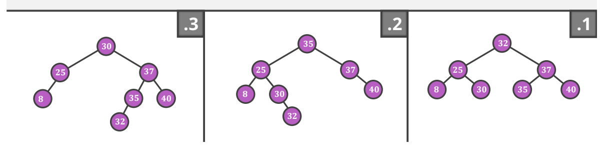
1. מחיקת צומת לפי אלגוריתם המחיקה של BST $O(\log n)$
2. נטפס במסלול מהאבא של הצומת שנמחק עד לשורש $O(\log n)$

- לכל צומת בדרך נעדכן את שדה הגובה (או balance factor) ונבדוק האם יש הפרה של תכונת ה-AVL
- הפרה של תכונת ה- AVL תתרחש אם גורם האיזון של הצומת הפך להיות 2 או 2−
- במידה וצומת v במסלול העלייה אינו מקיים את תכונת האיזון, יש לבצע רוטציה מתאימה
- פעולת הרוטציה יכולה להקטין את הגובה של תת-העץ שעל שורשו היא פועלת, ולכן אם גובה של תת עץ השתנה, יש להמשיך לעלות במסלול לשורש

זמן ריצה הכולל $T(n) = O(\log n)$

? **מהו העץ שמתקבל לאחר מחיקת צומת בעל מפתח 10 מהעץ שבציור?**





סיכום