

# מעבדה 8. נושא: עץ חיפוש בינארי מאוזן

תאריך הגשה: 11/07/2024 בשעה 23:00 (בזוגות)

**יש לקרוא היטב לפני תחילת העבודה !**

## מבוא:

במעבדה הנוכחית נממש עצי חיפוש בינאריים מאוזנים – עצי AVL.

עצי AVL הם עצי חיפוש בינאריים המקיימים את התכונות הבאות:

1. פעולות החיפוש, ההוספה והמחיקה יתבצעו בסדר גודל של  $O(\log N)$
2. העץ קל לתחזוקה

עצי AVL מקיימים את התנאי הבא שמבטיח את התכונות הנ"ל:

לכל צומת בעץ הערך המוחלט של הפרש בין הגבהים של שני תתי העצים הוא לכל היותר 1.

## התרגיל:

יש לכתוב את המחלקה המממשת עץ AVL.

```
public class AVLTree<T extends Comparable<T>>
```

המחלקה תירשם בתוך החבילה `il/ac/telhai/ds/trees`  
הסברים על אופן המימוש של האיזון - בסוף התרגיל.

עליכם לממש בנאי כדלהלן:

```
public AVLTree(T value)
```

ואת המתודות הבאות:

```
//add the value to the tree, and return the updated root of the tree.
```

```
public AVLTree<T> add(T value)
```

```
//return the value in this node
```

```
public T getValue()
```

```
//return the left subTree of this node
```

```
public AVLTree<T> getLeft()
```

```
//return the right subTree of this node
```

```
public AVLTree<T> getRight()
```

## הנחיות:

ניתן לעשות שימוש בקוד שרשמתם במעבדות קודמות, אך אין חובה לעשות זאת.

הערה: שימו לב שי `getLeft` ו-`getRight` אלו פעולות שעלולות לגרום לכך שהעץ לא יהיה מאוזן. עם זאת, עליכם לממש פעולות אלו והן חשובות לצרכי בדיקה.

# סדר העבודה ופרטים טכניים

שליפת הפרויקט DS-Lab08-AVLTree מתוך GITHUB בקישור:

<https://github.com/michalHorovitz/DSLAb2024Public>

- אם אין לכם גישה לפרויקט שהורדתם מ GITHUB במעבדה הראשונה יש לבצע שליפה מחדש לפי ההוראות במעבדה הראשונה.
  - אם יש לכם גישה לפרויקט שהורדתם מ GITHUB במעבדה הראשונה אז בצעו:
    - קליק על שם הפרויקט.
    - עכבר ימני
    - Team-->Pull
    - File-->Import->Git->Projects From Git->Existing Local Repository
- אם אתם עובדים ב VDI, מומלץ לשנות את המיקום המוצע לפרויקט בתיקייה כלשהי בכוון H.

## פורמט קובץ ההגשה ובדיקתו:

**פורמט:** יש להגיש קובץ ZIP בשם

51\_lab08\_123456789\_987654321.zip

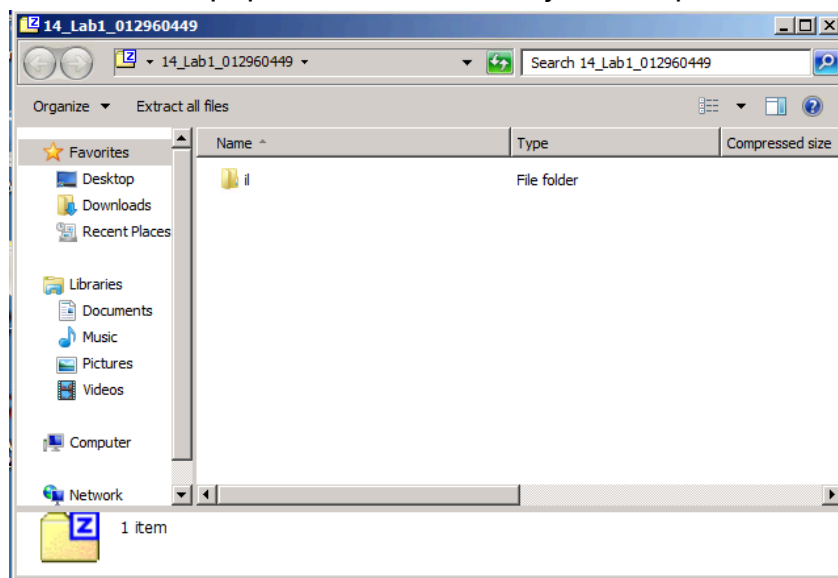
(כמובן, יש להחליף את המספרים עם מספרי ת.ז. של המגישים).

על הקובץ להכיל את כל קבצי ה JAVA שכתבתם כאשר הם נמצאים בתיקייה

il/ac/telhai/ds/trees

כלומר, השורש של קובץ ההגשה יכיל רק תיקייה בשם il.

ומכיל את כל קבצי - java . להמחשה תמונה של קובץ כזה שנפתח ב - WindowsExplorer



**בדיקת קובץ ההגשה:** בדקו את הקובץ שיצרתם בתוכנת הבדיקה בקישור:

<https://csweb.telhai.ac.il/>

ראו [סרטון הדגמה](#) של השימוש בתוכנת הבדיקה.

**חשוב !!!**

בדיקת ההגשות תבוצע ברובה ע"י תוכנית הבדיקה האוטומטית הנ"ל. תוצאת הבדיקה תהייה בעיקרון זהה לתוצאת הבדיקה הנ"ל שאתם אמורים לערוך בעצמכם. כלומר, אם ביצעתם את

הבדיקה באתר החוג, לא תקבלו הפתעות בדיעבד. אחרת, ייתכן שתרגיל שעבדתם עליו קשה ייפסל בגלל פורמט הגשה שגוי וכו'. דבר שהיה ניתן לתקנו בקלות אם הייתם מבצעים את הבדיקה. היות ואין הפתעות בדיעבד, לא תינתן אפשרות של תיקונים, הגשות חוזרות וכד'.

הגשה שלא מגיעה לשלב הקומפילציה תקבל ציון 0.

הגשה שלא מתקמפלת תקבל ציון נמוך מ- 40 לפי סוג הבעיה.

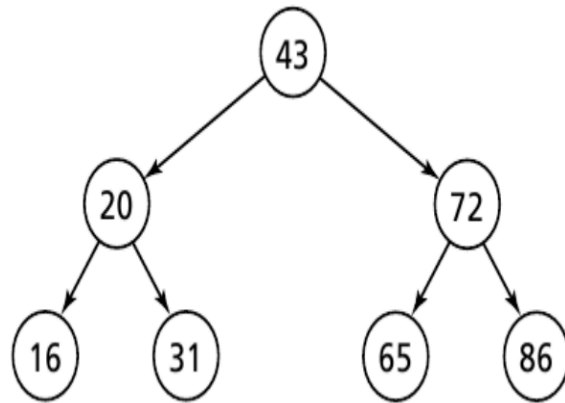
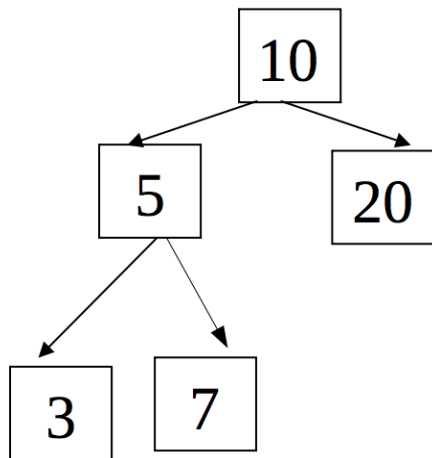
הגשה שמתקמפלת תקבל ציון 40 ומעלה בהתאם לתוצאות הריצה, ותוצאת הבדיקה הידנית של הקוד (חוץ ממקרה של העתקה).

תכנית הבדיקה האוטומטית מכילה תוכנה חכמה המגלה העתקות. מקרים של העתקות יטופלו בחומרה

# מימוש עצי AVL

**הגדרה 1 (עץ AVL):** עץ חיפוש בינארי בו הפרש הגבהים של תת-העץ השמאלי ותת-העץ הימני (לכל צומת פנימי) יהיה 1 לכל היותר. ייקרא עץ AVL על שם Adelson-Velskii and Landis.

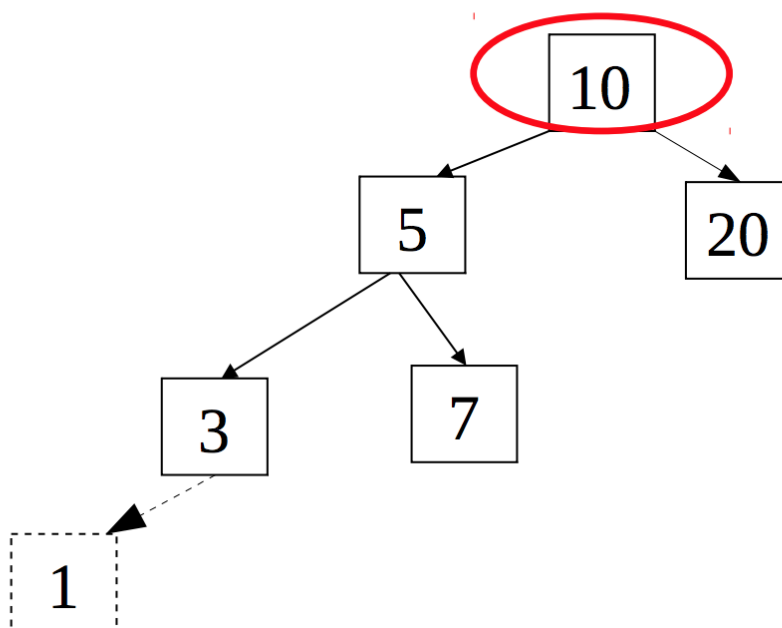
דוגמא לעצי AVL:



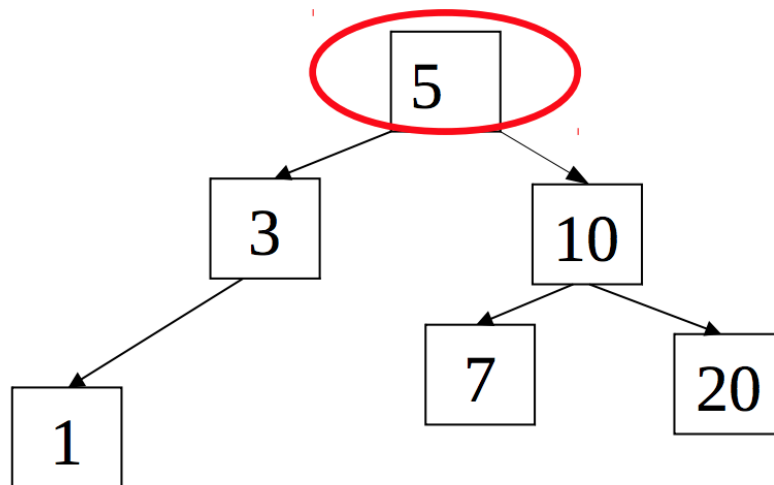
## הכנסת צומת לעץ AVL

ההכנסה נעשית באופן שגורתי לעצי חיפוש: הצומת החדש מתווסף בתור עלה. באופן כזה לא נפגעת תכונת החיפוש של העץ. אולם, הכנסה עלולה לפגוע בתנאי האיזון. נסמן ב-a את הצומת הראשון במסלול מנקודת ההכנסה אל השורש, שבו יש הפרה של תנאי האיזון.

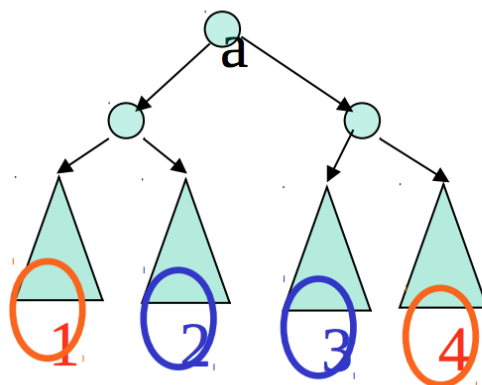
למשל, הכנסה של 1 גורמת להפרת האיזון בשורש (a הוא השורש):



עלינו לתקן את האיזון בין תתי העצים של צומת זה (נראה כי זה מספיק). ניתן לעשות זאת אם נהפוך את 5 להיות שורש העץ ואת 7 לבן השמאלי של 10. נקבל:



יש ארבעה מקרים, אך משיקולי סימטריה אפשר לדון רק בשני מקרים:  
 א. הכנסה "חיצונית". לתת-עץ שמאלי של בן שמאלי או לתת-עץ ימני של בן ימני. מקרים 1 ו-4  
 ב. הכנסה "פנימית". לתת-עץ ימני של בן שמאלי או לתת-עץ שמאלי של בן ימני. מקרים 2 ו-3



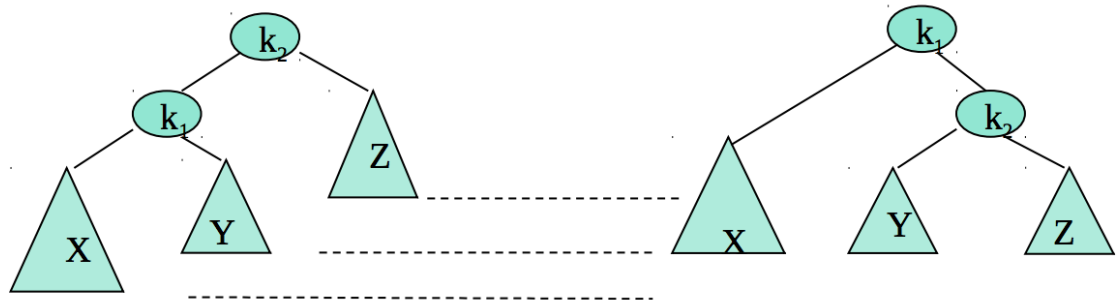
## פתרון הפרת האיזון:

כאמור נסמן ב-a את הצומת הראשון במסלול מנקודת ההכנסה אל השורש, שבו יש הפרה של תנאי האיזון. בהכנסה "חיצונית", (למשל: לתת-עץ השמאלי של הבן השמאלי של a) נראה שמספיק סיבוב אחד. בהכנסה "פנימית", (למשל: לתת-עץ הימני של הבן השמאלי של a) נראה שיש צורך בשני סיבובים.

### הכנסה חיצונית (מקרים 1 ו 4)

#### רוטציה יחידה

צד שמאל מתאר את המצב לפני התיקון (הרוטציה): אחרי ההוספה לתת-העץ X מופר האיזון. הצומת הראשון שאינו מאוזן מסומן ב-  $k_2$ . צד ימין מתאר את המצב לאחר התיקון.



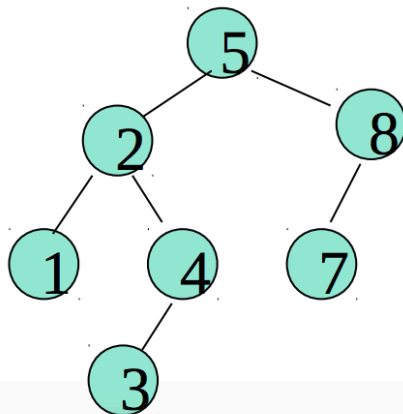
הערה: באיור לעיל הראנו רוטציה ימנית שפותרת את המקרה 1. תארו באיור דומה רוטציה שמאלה לפתרון מקרה 4.

#### תרגיל עצמי 1

הכנס מפתח 6 לתוך העץ הנתון.

יש להכניס את המפתח באופן רגיל ואז לבצע רוטציה יחידה.

מהם  $k_1$ ,  $k_2$ , X, Y, Z במקרה זה?



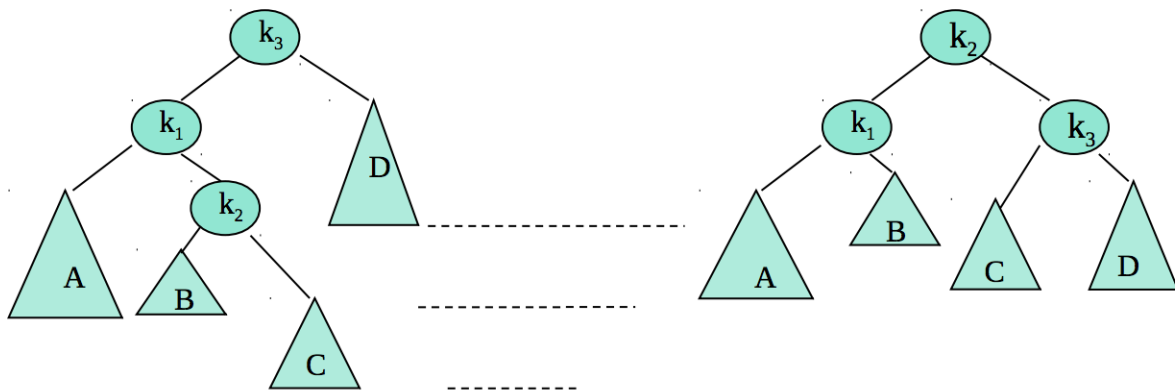
## תרגיל עצמי 2

בנה עץ AVL אשר נוצר ע"י הכנסת המפתחות הבאים (משמאל לימין):

3 2 1 4 5 6 7

הכנסה פנימית (מקרים 3 ו 2)

רומציה כפולה



מכניסים לתת-העץ הימני של הבן השמאלי, לבן הוא אינו ריק. לבן אפשר לתאר את העץ באמצעות 4 תתי-עצים (אולי ריקים) ועוד שלושה צמתים מקשרים.

$K_3$  הוא הצומת הראשון (מלמעלה) שבו מתגלה ההפרה. נניח בה"כ שהכנסנו ל-C ולכן הוא עמוק יותר.

1.  $k_3$  לא יכול להישאר השורש ורומציה עם  $k_1$  לא פותרת את הבעיה, לבן נהפוך את  $k_2$  להיות השורש.

2. הבנים שלו חייבים להיות: משמאל  $k_1$  מימין  $k_3$  המיקום של ארבעת תתי-העץ A עד D נקבע בהתאם.

3. עושים רומציה שמאלה בין  $k_2$  ל- $k_1$  ואח"כ רומציה ימינה בין  $k_2$  ל- $k_3$ .

4. זה נקראה רומציה שמאלה-ימינה.

הערה: המקרה האנלוגי – הכנסה לתת-עץ של הבן הימני נפתר ע"י רומציה ימינה-שמאלה.

## תרגיל עצמי 3

לעץ שיצרתם בתרגיל עצמי 2 הכניסו את המפתחות הבאים (משמאל לימין):

16,15,14,13,12,11,10,8,9

### אלגוריתם ההכנסה של צומת X לעץ AVL

1. אם העץ ריק, צור צומת חדש והכנס בו את X, השם 0 בשדה הגובה.
2. אחרת, אם X קטן מהמפתח של השורש, הכנס משמאל, בדוק אם יש הפרת איזון ומפל בה
3. אחרת אם X גדול או שווה מהמפתח של השורש, הכנס מימין, בדוק אם יש הפרת איזון ומפל בה
5. עדכן את גובה העץ והחזר אותו.

#### **טיפול בהפרת איזון:**

בהכנסה לתת-עץ שמאלי: אם X קטן מהמפתח של הבן השמאלי בצע סיבוב אחד, אחרת בצע סיבוב כפול.

בהכנסה לתת-עץ ימני: אם X גדול מהמפתח של הבן הימני בצע סיבוב אחד, אחרת בצע סיבוב כפול.