

תאריך: ____ . ____ . ____

מקצוע: מדעי המחשב ב' - JAVA

שם: _____

נושא: עץ בינרי

עץ בינרי – דף עבודה מס' 1

חלק מהתרגילים נלקחו מהחומר שהוכן על-ידי ולרי פקר משתתף קורס מורים מובילים תשע"ה

1. נתון הקובץ BuildTree.Java
2. הגדרי פרויקט בשם TreesProject.
3. הגדרי מחלקה בשם BuildTree.
4. העתקי את הקובץ BuildTree.Java למחלקה ב-Eclipse.
5. הרצי את התכנית הראשית.
6. הגדרי סדר גודל לתכנית הראשית, נמקי!

```
import unit4.collectionsLib.*;
import unit4.utilsLib.*;

public class BuildTree
{
    public static void main(String [] args)
    {
        BinTreeNode<Integer> t1 = new BinTreeNode<Integer>(3);
        BinTreeNode<Integer> t2 = new BinTreeNode<Integer>(3);
        BinTreeNode<Integer> t3 = new BinTreeNode<Integer>(t1, 6, t2);
        BinTreeUtils.showTree(t3, "first");

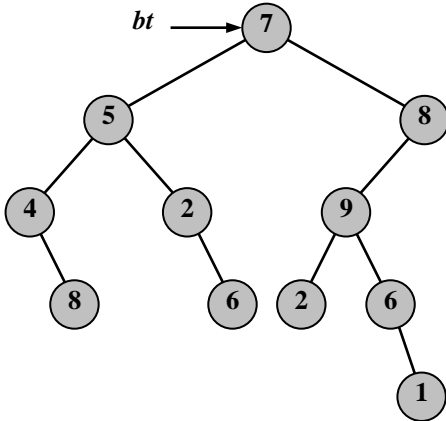
        BinTreeNode<Integer> t4 = new BinTreeNode<Integer>(4);
        BinTreeNode<Integer> t5 = new BinTreeNode<Integer>(null, 2, t4);
        BinTreeNode<Integer> t6 = new BinTreeNode<Integer>(t5, 2, null);
        BinTreeUtils.showTree(t6, "second");

        BinTreeNode<Integer> t = new BinTreeNode<Integer>
            (t3, 5, new BinTreeNode<Integer>(t6, 6, null));
        BinTreeUtils.showTree(t, "third");
    }
}
```



משימה 1 – חלק א' (בניית עץ חוליות בינרי בצורה ידנית)

תרשים העץ, שצריך לבנות:

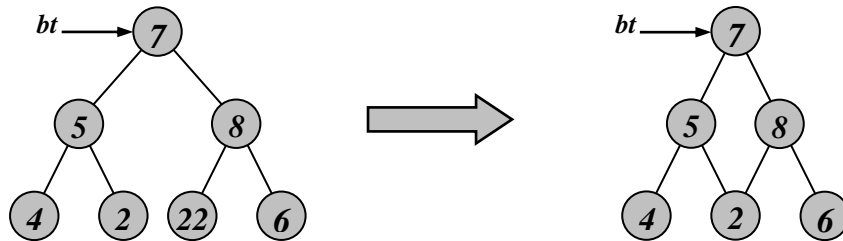


משימה 2

שינוי מבנה של עץ צריך לעשות **בזהירות רבה**. מבנה כזה חייב לקיים את תנאי ההגדרות של "עץ בינרי". קל לקלקל את המבנה, כך שיהפוך למבנה שאינו עץ חוליות בינרי, תוך שימוש בפעולות `setLeft()` ו-`setRight()`.

בדוגמה שלהלן המבנה, שאינו עץ חוליות בינרי, בצד ימין יתקבל מהעץ מצד שמאל על-ידי ההוראות:

```
bt.getRight().setLeft( bt.getLeft().getRight() );
```

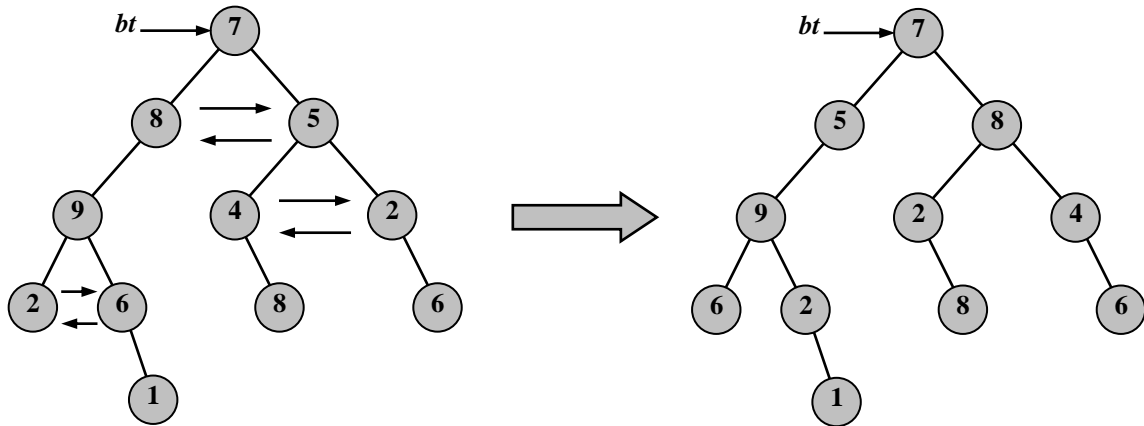


- א. בצעי את השינוי במחשב, והרץ/י את התכנית שוב.
האם בפלט הגרפי של המחשב קבלת את אותו עץ ימני הנ"ל? מדוע?
- ב. הוסף/י את השורה הבאה:
`bt.getRight().getLeft().setInfo(3);`
בצעי את השינוי במחשב, והרץ/י את התכנית שוב.
- ג. הוסף/י את השורה הבאה:
`bt.getRight().getLeft().getRight().setInfo(10);`
בצעי את השינוי במחשב, והרץ/י את התכנית שוב.



משימה 3 – (שינוי ערכים של עץ חוליות בינרי)

הפעולה, שאתם צריכים לכתובת, היא פעולה רקורסיבית, שאמורה להחליף את ערכי האחים בעץ בינרי בלי לשנות את מבנהו. הפעולה מחליפה את ערכי האחים בעץ כמו התרשים הבאה:



לפניכם מימוש הפעולה `swapBrothers`. בפעולה חסרות הוראות אחדות. השלימו את ההוראות החסרות לפי התיאור הרקורסיבי.

```

static void swapBrothers ( BinTreeNode < Integer > bt )
{
    // הפניה לילד השמאלי של העץ bt, שהוא שורש תת-העץ השמאלי
    BinTreeNode < Integer > tLeft = bt.getLeft() ;
    // הפניה לילד הימני של העץ bt, שהוא שורש תת-העץ הימני
    BinTreeNode < Integer > tRight = bt.getRight() ;
    // כלל עצירה : עץ bt הוא עץ עלה (העץ הקטן ביותר)
    if (( tLeft == null ) && ( tRight == null ))
        System.out.print ( "" ) ;
    // המקרה הכללי הראשון : שורש עם שני ילדים (שמאלי וימני)
    else
        if (( tLeft != null ) && ( tRight != null ))
        {
            // החלפת ערכי האחים של השורש bt
            _____
            _____
        }
}

```



```

_____  

//      כאן יופיע זימון רקורסיבי של הפעולה שמחליפה את ערכי כל האחים  

//      בתת-העץ השמאלי tLeft  

_____  

//      כאן יופיע זימון רקורסיבי של הפעולה שמחליפה את ערכי כל האחים  

//      בתת-העץ הימני tRight  

_____  

}  

else  

//      המקרה הכללי השני : שורש רק עם ילד שמאלי (ואין ילד ימני)  

if ( tLeft != null )  

{  

//      כאן יופיע זימון רקורסיבי של הפעולה שמחליפה את ערכי כל האחים  

//      בתת-העץ השמאלי tLeft  

_____  

}  

else  

//      המקרה הכללי השלישי : שורש רק עם ילד ימני (ואין ילד שמאלי)  

//      כאן יופיע זימון רקורסיבי של הפעולה שמחליפה את ערכי כל האחים  

//      בתת-העץ הימני tRight  

_____  

} // swapBrothers

```

