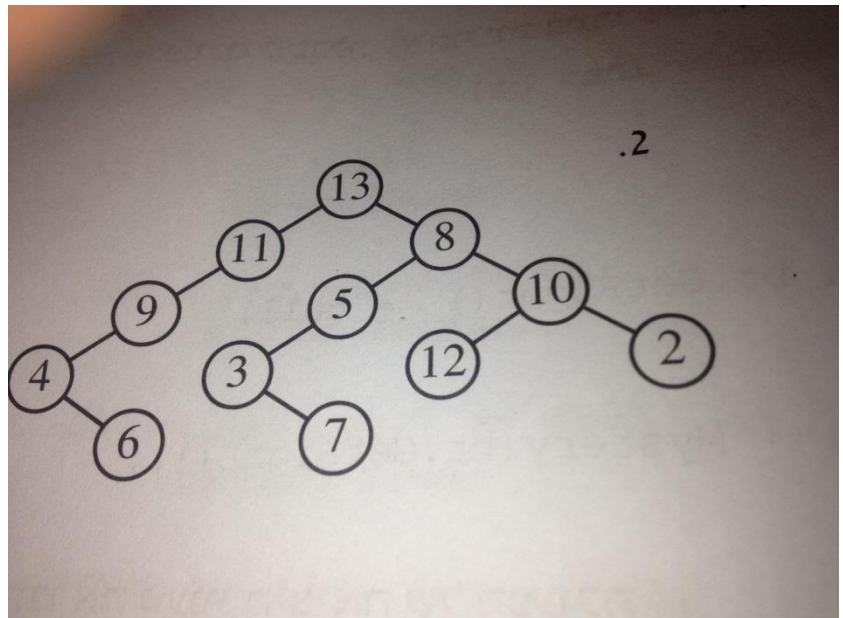


עץ בינרי

הוא המבנה שקיימים צמתים.

צומת אחד הוא שורש העץ ולכל צומת בעץ יש לכל היותר שני בנים.

דוגמה:



מושגים:

שורש העץ

אח

גובה העץ

צאצא

בן שמאלי

בן ימני

עלה

עץ מלא

רמה של עץ

תת עץ

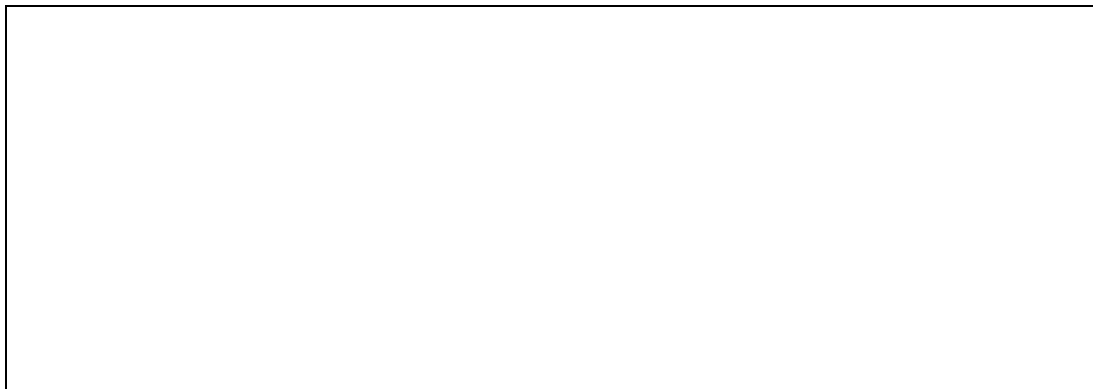
ניתן להרחיב את המושגים בעמודים 184 עד 186 מהספר

משימה:

סמנו בעץ של הדוגמה:

- (א) את העלים
- (ב) את האחים
- (ג) את גובה העץ
- (ד) את הצאצאים של 8
- (ה) את התת עץ המתחיל ב 11

דוגמאות מציאותיות לעץ בינרי:



המחלקה עץ חוליות בינרי

לייצוג עץ בינרי נשתמש בחוליות עם שתי הפניות. חוליה כזו נקראת חוליה בינרית `BinNode`. החוליה הבינרית תשמש לייצוג צומת בעץ.

לחוליה כזו יש שלוש תכונות: תכונה המכילה את הערך השמור בצומת ושתי תכונות שהן הפניות לעצמים נוספים מטיפוס `BinNode`. כאשר ערך ההפניה הוא `null` פירוש הדבר שהתת-עץ המתאים אינו קיים.

למעשה, עץ חוליות בינרי הוא בנוי בהפניה לחוליה הראשונה (שורש העץ) שממנו יוצאים הפניות נוספות (דומה לשרשרת חוליות).

מחלקת <code>BinNode<T></code>	
מייצגת חוליה בינרית מטיפוס <code>BinNode<T></code> שמכילה ערך מטיפוס <code>T</code> והפניות לשתי חוליות בינאריות. ניתן להשתמש במחלקה זאת כדי לייצג עץ בינרי המורכב מאפס או יותר חוליות בינאריות.	
בנאים:	
<code>BinNode (T x)</code>	• בונה חוליה בינרית: משים (מלשון השמה) את ערך החוליה ל- <code>x</code> , ואת שתי ההפניות שלה ל- <code>null</code>
<code>BinNode (BinNode<T> left, T x, BinNode<T> right)</code>	• בונה חוליה בינרית: משים את ערך החוליה ל- <code>x</code> , ואת שתי ההפניות שלה לחוליות הבינאריות <code>left</code> ו- <code>right</code> ערך של כל אחת משתי ההפניות הללו יכול להיות <code>null</code>
שאלות:	
<code>T getValue()</code>	• מחזיר את ערך החוליה
<code>BinNode<T> getLeft()</code>	• מחזיר את החוליה השמאלית, או <code>null</code>
<code>BinNode<T> getRight()</code>	• מחזיר את החוליה השמאלית, או <code>null</code>
<code>String toString()</code>	• מחזיר את ערך החוליה כמחרוזת
פקודות:	
<code>void setValue (T x)</code>	• משנה את ערך החוליה ל- <code>x</code>
<code>void setLeft (BinNode<T> left)</code>	• משים את ההפניה לחוליה השמאלית ל- <code>left</code> ערכו של <code>left</code> יכול להיות <code>null</code>
<code>void setRight (BinNode<T> right)</code>	• משים את ההפניה לחוליה הימנית ל- <code>right</code> ערכו של <code>right</code> יכול להיות <code>null</code>

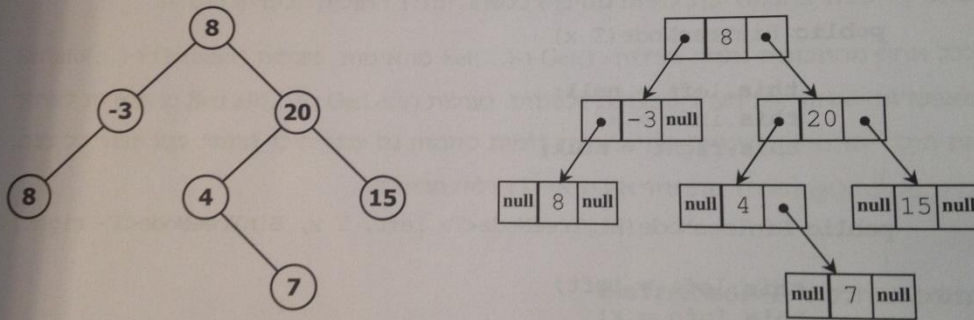
```

public class BinNode<T>
{
    private T Value;
    private BinTreeNode<T> left;
    private BinTreeNode<T> right;
    public BinNode(T value)
    {
        this.value = value;
        this.left = null;
        this.right = null;
    }
    public BinNode(BinNode<T> left, T value, BinNode<T> right)
    {
        this.value = value;
        this.left = left;
        this.right = right;
    }
    public T getvalue()
    {
        return value;
    }
    public BinNode<T> getLeft()
    {
        return left;
    }
    public BinNode<T> getRight()
    {
        return right;
    }
    public void Setvalue(T value)
    {
        this.value = value;
    }
    public void SetLeft(BinNode<T> left)
    {
        this.left = left;
    }
    public void SetRight(BinNode<T> right)
    {
        this.right = right;
    }
    public override string ToString()
    {
        return " ( " + left + " " + info + " " + right + " ) ";
    }
}

```

ד. עץ חוליות בינרי

המחלקה BinTreeNode מאפשרת ליצור חוליות בינריות, לחבר אותן זו לזו באמצעות הפעולות SetLeft(...) ו-SetRight(...), וכך לבנות מבני חוליות היררכיים שייצגו עצים בינריים. לדוגמה, מבנה החוליות הבינריות מימין מייצג את העץ הבינרי המופיע משמאל:



לשם הפשטות, במקום הביטוי המדויק אך המסובך: "מבנה חוליות המייצג עץ בינרי", נשתמש מכאן והלאה בביטוי הפשוט יותר: "עץ חוליות בינרי", ואף בביטוי המקוצר: "עץ בינרי". כמו כן, נשתמש לעתים קרובות במונח "צומת" במקום במונח "חוליה".

הגדרה רקורסיבית של עץ חוליות בינרי:

עץ חוליות בינרי הוא:

חוליה בינרית יחידה

או

חוליה בינרית שבה יש הפניה אחת לעץ חוליות בינרי או שתי הפניות לעצי חוליות בינריים הזרים זה לזה (שאינן להם חוליות משותפות)

שימו לב: לא קיים עץ בינרי ריק!

הבדיקות אם העץ ריק הן רק לצורך בניית הפעולות הרקורסיביות

שימוש במחלקה BinNode<T> - קובץ TreeNision

ציורים של עצים - תרגול בסוף החוברת

פעולות דוגמאות:

דוגמה 1

הדפסת ערכי העץ:

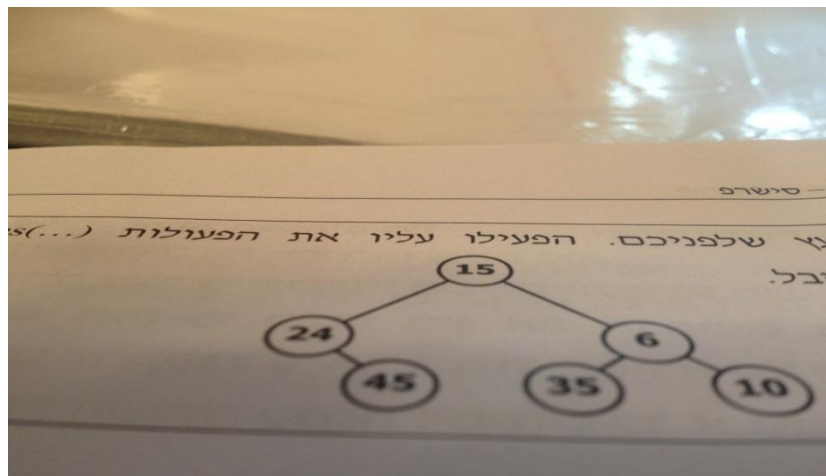
```
public static void PrintNodes(BinNode<int>bt)
{
    if(bt!=null)
    {
        Console.WriteLine("{0}",bt.GetValue());

        PrintNodes(bt.getLeft());

        PrintNodes(bt.getRight());
    }
}
```

משימה:

כתוב טבלה רקורסיבית עבור הדוגמה 1 ועץ הבא:



דוגמה 2

ספירת מספר הצמתים בעץ:

```
public static int NumNodes(BinNode<int>bt)
{
    if(bt==null)
        return 0;

    return NumNodes(bt.getLeft())+NumNodes(bt.getRight())+1;
}
```

}

תרגילים

שאלה 1

נתונה הפעולה הבאה:

```
public static int Mystery(BinNode<int> bt)
{
    if (bt==null)
        return 0;
    if(bt.getLeft()==null && bt.getRight()==null)
        return 1;
    return Mystery(bt.getLeft()+Mystery(bt.getRight()));
}
```

- (א) כתבו את טענת היציאה של הפעולה.
(ב) אם היינו מחליפים את טיפוס ערכי הצמתים מ int לכל טיפוס אחר, הדבר לא היה משפיע על ביצוע הפעולה, והיא הייתה מבצעת בדיוק אותה משימה. האם טענה זו נכונה? נמק!

שאלה 2

נתונה הפעולה הבאה:

```
public static bool secret(BinNode<char>bt)
{
    if(bt.getLeft()==null && bt.getRight()==null)
        return true;
    if(bt.getLeft()==null || bt.getRight()==null)
        return false;
    return secret(bt.getLeft()) && secret(bt.getRight());
}
```

- מה מבצעת הפעולה?
(המלצה: לבנות טבלת מעקב.....)

שאלה 3

לפניך הפעולה הבאה:

```
Public static bool What(BinNode<int>bt,int x)
{
    if (bt==null)
        return false;
    if (bt.GetValue()!=x)
        return false;
    if(bt.GetLeft()==null && bt.GetRigth()==null)
        return true;
    return (what(bt.GeLeft(),x) || what(bt.GetRigth(),x))
}
```

}

- (א) שרטט 2 עצים כך שהפעולה תחזיר עבורם TRUE.
(ב) שרטט 2 עצים כך שהפעולה תחזיר עבורם FALSE.
(ג) מה הפעולה מבצעת?

שאלה 4

לפניכם פעולה:

```
public static BinNode<int>Build(int n)
{
    if(n==0)
        return new BinNode<int>(0);
    return(new BinNode<int>(Build(n-1),n,Build(n-1)));
}
```

- (א) ציירו את העץ הבינרי שיוחזר עבור הזימון Build(3)
(ב) מה מבצעת הפעולה Build באופן כללי עבור כל מספר n גדול או שווה ל 0?

שאלה 5

לפניכם הפעולה:

```
public static int Sod(BinNode<int>bt)
{
    int ml=bt.getValue();
    int mr=bt.getValue();
    if(bt.getLeft()!=null)
        ml=Math.Max(bt.getValue(),Sod(bt.getLeft()));
    if(bt.getRight()!=null)
        mr=Math.Max(bt.getValue(),Sod(bt.getRight()));
    return Math.Max(ml,mr);
}
```

מה היא מבצעת?

שאלה 6

(א)

ממשו את הפעולה הבאה:

```
public static int count(BinNode<char>bt,char ch)
```

הפעולה תחזיר את מספר המופעים של התו ch בעץ bt.

(ב)

כתבו פעולה המקבל עץ בינרי של תווים ו תו וחזירה "אמת" אם התו מופיע בעץ או "שקר" אחרת.

שאלה 7

כתוב פעולה המקבלת עץ בינרי של מספרים שלמים ומחזירה את גבוהו.

שאלה 8

ממשו את הפעולה:

```
Public static BinNode<int>BuildIdent(BinNode<int>bt)
```

הפעולה תבנה עץ בינרי זהה (במבנה ובתוכן) לעץ bt המקבל כפרמטר ותחזיר אותו.

שאלה 9

כתוב פעולה המקבלת עץ בינרי של מספרים שלמים ומחזירה "אמת" אם העץ הוא "עץ בינרי מלא" ו"שקר" אחרת.

(עץ מלא הוא העץ שיש לכל צומת בדיוק 2 בנים או שהוא עלה..)

שאלה 10 (tree10)

עצים דומים הם עצים שהמבנה הפנימי שלהם, כלומר סדר הצמתים בתוך כל עץ, זהה, אך לא בהכרח מכילים את אותם איברים.

(א) כתוב זוג עצים דומים.
(ב) ממש את הפעולה sametree המקבלת 2 עצים של מספרים שלמים ומחזירה "אמת" אם שני העצים הם "עצים דומים" ו"שקר" אחרת.

מעברים על עץ בינרי

(א) סריקה תחילית:

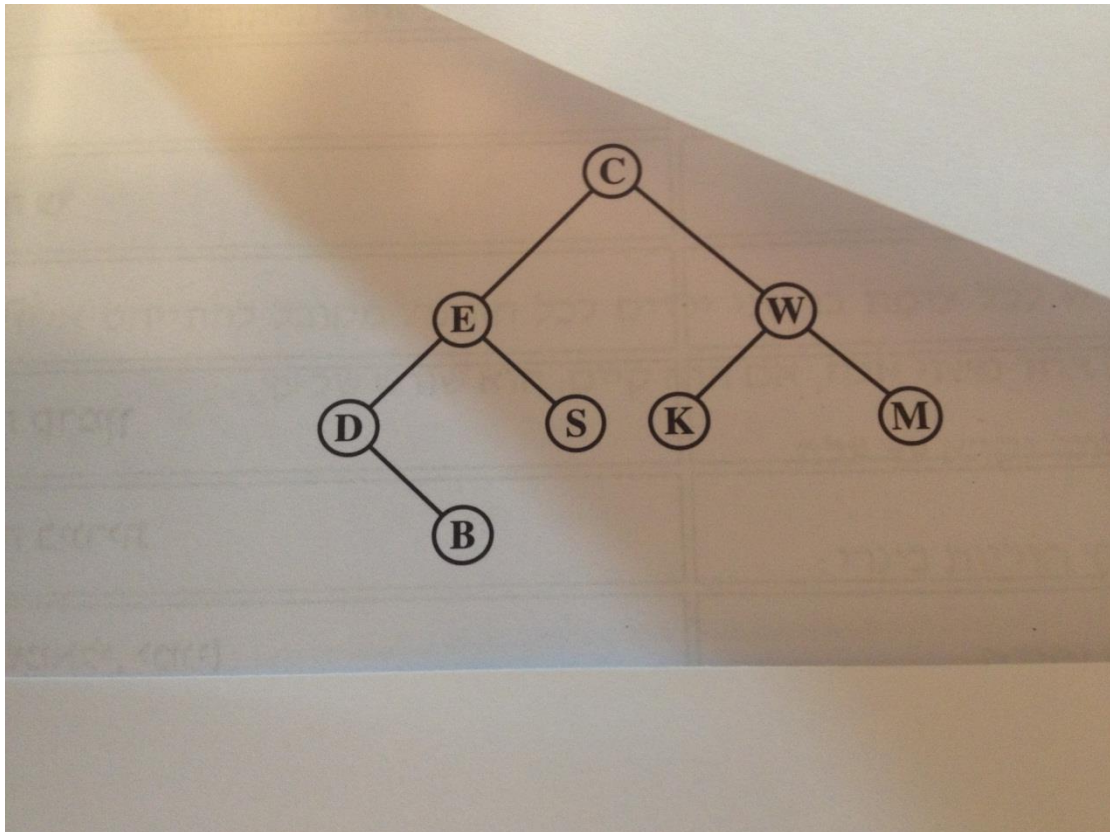
ביקור בשורש

סריקה רקורסיבית בשמאל

סריקה רקורסיבית בימין

כלל: נבקר בשורש כאשר נפגוש אותו בפעם הראשונה (ולפי הכלל שמאל ימין)

דוגמה: עבור העץ הבא:



ועבור הפעולה הבאה:

```
public static void PrintNodes(BinNode<int>bt)
{
    if(bt!=null)
    {
        Console.Write("{0}",bt.GetValue() + "");
        PrintNodes(bt.getLeft());
        PrintNodes(bt.getRight());
    }
}
```

בסריקה תחילית נקבל את סדר הצמתים הבא:

(ב) סריקה תוכית:

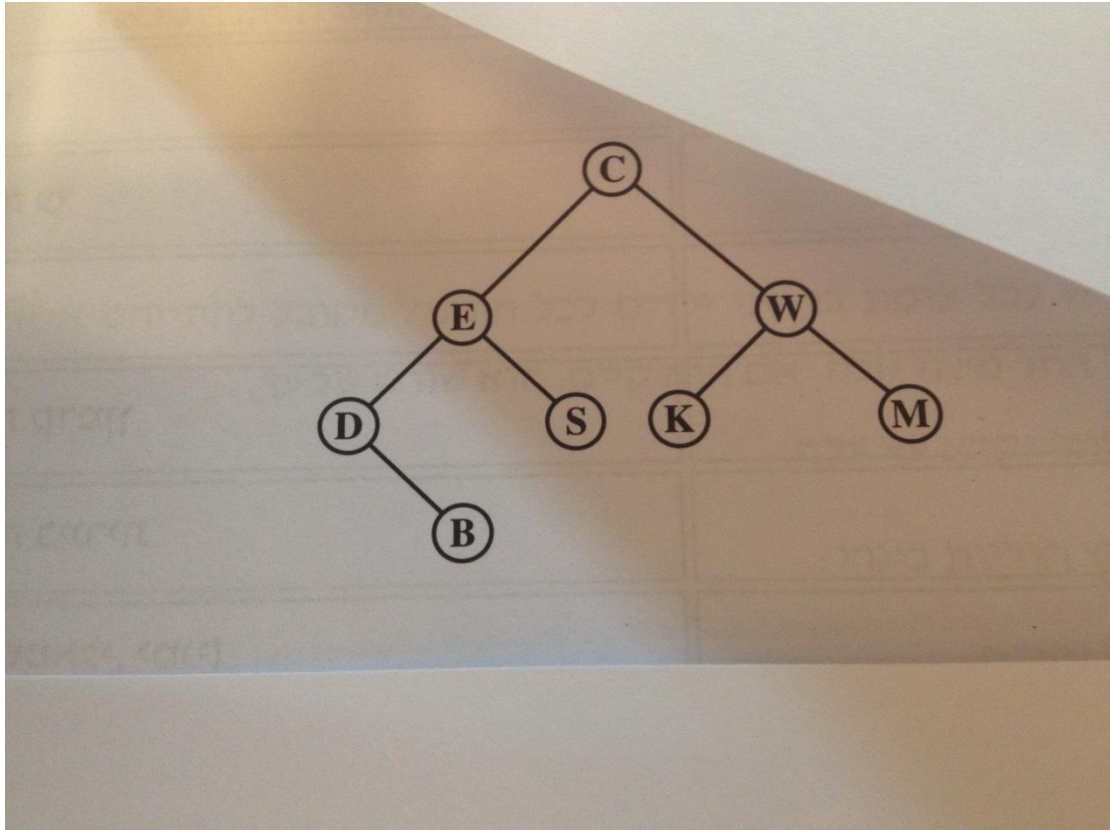
סריקה רקורסיבית בשמאל

ביקור בשורש

סריקה רקורסיבית בימין

כלל: נבקר בשורש לפני שפונים ימינה

דוגמה: עבור העץ הבא:



ועבור הפעולה הבאה:

```
public static void PrintNodes(BinNode<int>bt)
{
    if(bt!=null)
    {
        PrintNodes(bt.getLeft());
        Console.Write("{0}",bt.GetValue() + "");
        PrintNodes(bt.getRight());
    }
}
```

בסריקה תוכית נקבל את סדר הצמתים הבא:

ג) סריקה סופית:

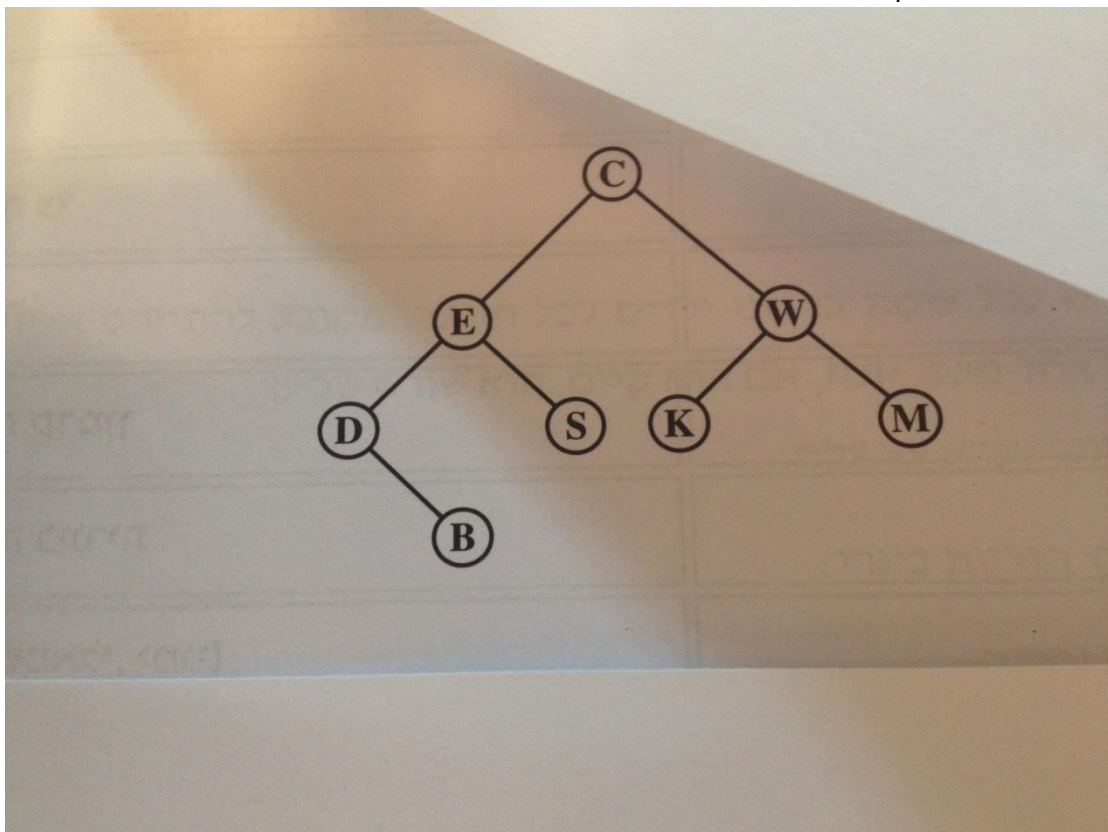
סריקה רקורסיבית בשמאל

סריקה רקורסיבית בימין

ביקור בשורש

כלל: נבקר בשורש כאשר נפגוש אותו בפעם האחרונה (ולפי הכלל שמאל ימין)

דוגמה: עבור העץ הבא:



ועבור הפעולה הבאה:

```
public static void PrintNodes(BinNode<int>bt)
{
    if(bt!=null)
    {
        PrintNodes(bt.getLeft());
        PrintNodes(bt.getRight());
        Console.Write("{0}",bt.GetValue() + "");
    }
}
```

בסריקה סופית נקבל את סדר הצמתים הבא:

--

תרגילים (המשך)

שאלה 11

ציירו שני עצים שונים כך שאם נסרוק אותם בשיטה תחילית נקבל את הסדרה: 3,7,9,5 (משמאל לימין).

שאלה 12

כאשר נתונים ערכים שהתקבלו משתי סריקות: בסדר תחילי ובסדר סופי, ייתכנו כמה עצים המתאימים לשתי הסריקות. ציירו שני עצים שונים ובהם ערכים, כך שאם נסרוק אותם בסדר תחילי נקבל את הסדרה: 3,7,9,5 ואם נסרוק אותם בסדר סופי נקבל 9,5,7,3 (תמיד משמל לימין)

שאלה 13

כאשר נותנים ערכים שהתקבלו משתי סריקות, ואחת מהן היא סריקה בסדר תוכי, קיים רק **עץ אחד** המתאים לשתי הסריקות (בתנאי שהמספרים בכל סריקה שונים לחלוטין זה מזה).

ציירו את העץ היחיד האפשרי שאם נסרוק אותו בסדר תחילי נקבל את הסדרה :

1,5,7,3,8,4,6

ואם נסרוק אותו בסדר תוכי נקבל: 7,5,8,3,1,4,6

שאלה 14

כתוב פעולה המקבלת עץ בינרי של מספרים שלמים ומעדכנת את העץ כך שכל מספר זוגי בו יהיה מחצית ממנו.

שאלה 15 (tree15)

כתוב פעולה המקבלת עץ בינרי של מספרים שלמים ומוסיפה אח לכל צומת שהוא בן יחיד. ערך האח הוא אפס.

המלצה: כדאי לבנות פעולת עזר בשם kodChild הבודקת אם לצומת יש 0 בנים או בן ימין בלבד או בן שמאל בלבד או שני בנים. פעולה זו אינה רקורסיבית ופועלת על צומת אחד בלבד.

שאלה 16

עץ חילקיהו הוא:

עץ עלה

או

צומת שיש לו 2 בנים, שערכו של הבן השמאלי מתחלק ללא שארית בערכו של הבן הימני, ומנת החלוקה שלהם שווה לערכו של אביהם וכן אחד מבניו הוא "עץ חילקיהו".

- (א) צייר עץ בעל 5 צמתים לפחות שהוא "עץ חילקיהו".
(ב) כתוב פעולה המקבלת עץ בינרי של מספרים שלמים ומחזירה "אמת" אם הוא "עץ חילקיהו" ו"שקר" אחרת.

שאלה 17

כתוב פעולה המקבלת עץ בנארי של תווים ומחזירה רשימה מטיפוס PAIR כאשר עצם כזה מורכב מתו ומספר הפעמים שהוא מופיע בעץ.

שאלה 18 treehoberetpage13tar18

עץ מאיר הוא עץ עלה או שורש ושני בנים שהפרש הערכים ביניהם הוא לכל היותר 2 וגם **שכל אחד מהם הוא עץ מאיר**.

- (א) צייר עץ מאיר בעל 7 צמתים לפחות.
(ב) כתוב פעולה הבודקת אם עץ מסוים הוא עץ מאיר.

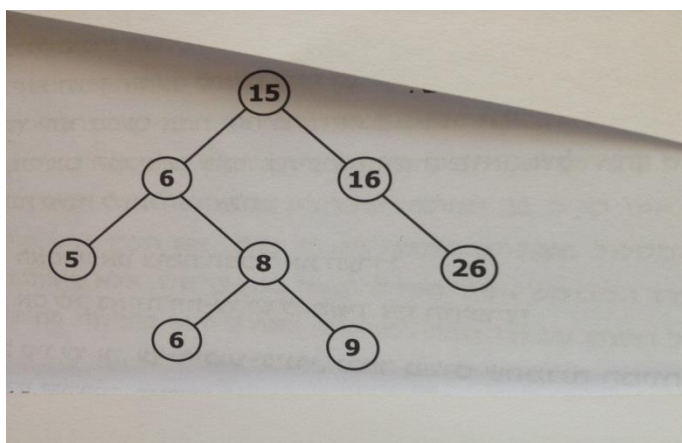
מומלץ לפתור את התרגילים בעמוד 16: "תרגילים סריקה"

שאלה 19

- (א) כתוב פעולה המקבלת עץ בינארי לא ריק של תווים ותו X . הפעולה תחזיר "אמת" אם קיים מסלול מהשורש עד עלה כלשהו שכל הצמתים בו הם התו X , או "שקר" אחרת.
(ב) מהי סיבוכיות זמן הריצה של הפעולה שכתבת בסעיף א'. נמק.!!
d:/TreMalula

עץ חיפוש בינרי

עץ חיפוש בינרי הוא העץ שכל שורש בו הוא גדול מכל הצמתים הנמצאים בתת-עץ השמאלי שלו וקטן או שווה מכל הצמתים הנמצאים בצד ימין שלו.



דוגמה:

היתרון בעץ חיפוש הוא שיעיל יותר למצוא ערך אם הוא נמצא בעץ!!!!

למה?

כתוב פעולה המקבלת עץ חיפוש בינרי ומספר ומחזירה "אמת" אם המספר נמצא בעץ חיפוש ו"שקר" אחרת.

פתרון

```
public static bool Existinb(BinNode<int>t,int x)
{
    if(t==null)
        return false;
    if(t.getValue()==x)
        return true;
    if(t.getValue()<x)
        return Existinb(t.getRight(),x)
    return Existinb(t.getRight(),x);
}
```

הערך הקטן ביותר בעץ חיפוש נמצא בצומת השמאלי ביותר בעץ

תרגיל

בנה עץ חיפוש בינרי לפי הסדרות הבאות:

(א) 5 6 8 9 15 26 29 30
(ב) 30 29 6 8 5 26 9 15
(ג) 8 29 9 30 26 5 6 15
(ד) 6 9 8 29 5 15 30 26

סריקה לפי רמות

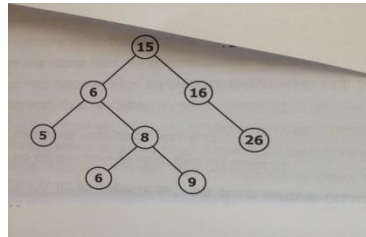
רמה של צומת מסוים בעץ היא אורך המסלול מהשורש אל צומת זה, כלומר המרחק של הצומת מהשורש.

רמת השורש היא 0. הרמה של כל צומת אחר בעץ גדולה באחד מהרמה של ההורה שלו.

גובה העץ זו הרמה הגבוהה ביותר של העץ.

נתון העץ הבא:

שרטט את הרמת שלו וקבע מהו הגובה שלו.



הסריקה לפי רמות היא פעולה שאינה רקורסיבית. סריקה זו משתמשת בתור כדי לבצע את הפעולה.

פתרון:

```
Public static void PrintMyTree(BinNode<int>tr)
{
    Queue<BinNode<int>>q=new Queue<BinNode<int>>();
    q.insert(tr);
    while(!q.IsEmpty())
    {
        BinNode<int >mytre=q.Remove();
        Console.Write("{0} ",mytre().GetValue());
        if(mytre.GetLeft()!=null)
            q.insert(mytre.GetLeft());
        if(mytre.GetRight()!=null)
            q.insert(mytre.GetRight());
    }
}
```

תרגיל

בנה טבלת מעקב עבור העץ הנתון ותור כלשהו q והצג את הפלט.

תרגילים סריקה לרוחב(לפי רמות)

עמוד 184 תרגילים 37, 38

תרגילים נוספים מהספר חזרות

תרגילים משולבים כל הנושאים

שאלה 1

המחלקה RunMan מייצגת נתונים עבור ספורטאי שרץ במהלך חודש אחד.
המחלקה מיוצגת ע"י שם הרץ, גיל וסה"כ ק"מ שהספורטאי רץ במהלך חודש אחד.

המחלקה GroupRunningCity מיוצגת ע"י קבוצה של רצים בעיר מסוימת ושם העיר.

- (א) כתוב את כותרת המחלקה RunMan ואת התכונות שלה.
 - (ב) כתוב את כותרת המחלקה GroupRuningCity ואת התכונות שלה.
 - (ג) כתוב פעולה חיצונית המקבלת עץ שכל צומת הוא מטיפוס GroupRunningCity בשם Myprinting המציגה את שמות הערים שממוצע הגילאים של הרצים בעיר שווה לממוצע הק"מ של אותה הקבוצה.
 - (ד) מהי סיבוכיות זמן הריצה של הפעולה שכתבת בסעיף ד. נמק.
- הערה: ניתן להניח שקיימות את כל הפעולות Get בכל המחלקות.

שאלה 2

כתוב פעולה חיצונית המקבלת מערך של רשימות שכל רשימה מורכבת מעצים של מספרים שלמים.

הפעולה תחזיר "אמת" אם כל עצים בכל הרשימות במערך הן עצים "מלאים" ו"שקר" אחרת.

עץ מלא הוא העץ שלכל צומת יש 2 בנים או שהוא עלה.

תרגילים נוספים על עצים

לכל השאלות יש לרשום סיבוכיות זמן ריצה

תרגילים סריקה.

שאלה 1

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תדפיס את הצמתים בעץ שיש להם מספרים קטנים מ 10 באופן תוכי.

שאלה 2

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תדפיס את הבנים הימניים בעץ באופן סופי.

שאלה 3

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר את מספר העלים השליליים בעזרת סריקה תוכית.

שאלה 4

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תדפיס את כל הצמתים שערכם נמוך מערכו של אביהם בעץ באופן סופי.

שאלה 5

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תדפיס את כל הצמתים שיש להם 2 בנים וערכם גדול מערך של לפחות אחד מהם. הסריקה באופן תוכי.

תרגילים ספירה/סכימה

שאלה 6

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר את סכום הצמתים בו.

שאלה 7

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר את כמות העלים שלו.

שאלה 8 TreePage17Tar8

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר את סכום הבנים הימניים שלו.

שאלה 9 tar7

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר את כמות הבנים היחידים שלו.

שאלה 10

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר את כמות הצמתים המכילים ערכים זוגיים בעץ.

שאלה 11

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם ומספר שלם נוסף X. הפעולה תחזיר את כמות הצמתים שערכם שווה ל-X.

שאלה 12 treepage17Targuil12

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר את כמות הסבים בעץ.

שאלה 13 treepage17targuil13

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר את סכום הצמתים היחידים שערכם אי זוגי.

שאלה 14

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר את כמות הצמתים שיש להם 2 בנים וערכם של שני הבנים זהה..

שאלה 15

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר את כמות הצמתים שקטנים לפחות מאחד מבניהם.

שאלה 16 treepage17targuil16

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר את כמות הצמתים שערכם גדול מהערך של אביהם. (אין צורך להתייחס לשורש שעץ)

שאלה 17

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם ומספר רמה L שלם. הפעולה תחזיר את כמות הצמתים ברמה L.

שאלה 18

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם ומספר רמה L שלם. הפעולה תחזיר את סכום הצמתים עד לרמה L (לא כולל רמה L).

שאלה 19

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם ומספר רמה L שלם. הפעולה תחזיר את כמות הבנים היחידים החל מרמה L ועד לגובה העץ (כולל רמה L).

שאלה 20

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם ומספרי רמה L1 ו L2 שלמים. הפעולה תחזיר את כמות הצמתים החל מרמה L1 ועד לרמה L2 (כולל L1 וכולל L2).

תרגילים שונים

שאלה 21

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר את גובה העץ.

שאלה 22

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר את המספר המקסימאלי בעץ.

תרגילים אמת/שקר

שאלה 23 treePage18Targuil23

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר אמת אם העץ מכיל אך ורק ערכים אי זוגיים. ושקר אם לא.

שאלה 24 Treepage18Targuil24

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר אמת אם כל זוג אחים בעץ הינם בעלי ערך זהה. ושקר אם לא.

שאלה 25 Treepage18Targuil25

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר אמת אם ערכם של כל העלים בעץ זהה לערך של אביהם. ושקר אם לא.

שאלה 26

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר אמת אם העץ אינו מכיל את המספר 8. ושקר אם לא.

שאלה 27 Treepage18Targuil27

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר אמת אם יש בעץ צומת שיש לו שני בנים וערכם של שני הבנים שונה. ושקר אם לא.

שאלה 28

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר אמת אם הצמתים הימניים בהם מכילים ערכים קטנים מאביהם והצמתים השמאליים בעץ מכילים ערך גדול מאביהם. ושקר אם לא.

שאלה 29

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם ומספר רמה L. הפעולה תחזיר אמת אם כל הצמתים החל מרמה L והלאה מכילים ערכים אי זוגיים בלבד. ושקר אם לא.

שאלה 30

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם ומספר רמה L. הפעולה תחזיר אמת אם כל הצמתים ברמה L הינם בנים יחידים וגם אבות לבנים יחידים. ושקר אם לא.

תרגילים התאמת עץ להגדרה

שאלה 31

עץ אריאל הינו עלה או עץ ששני בניו הינם עצי אריאל כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר אמת אם העץ הינו עץ אריאל. ושקר אם לא.

שאלה 32 Treepage18Targuil31

עץ פיני הינו עץ ריק או עץ שערכו של לפחות אחד מבניו גדול ממנו ולפחות אחד מבניו הינו עץ פיני. כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר אמת אם העץ הינו עץ פיני. ושקר אם לא.

שאלה 33

עץ יחיאל הינו עלה או עץ שערך שורשו שווה לסכום ערכי בניו, וגם כל אחד מבניו הינו עץ יחיאל. כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר אמת אם העץ הינו עץ יחיאל. ושקר אם לא.

שאלה 34 Treepage18Targuil34

עץ בינארי יקרא צחי אם הוא עלה או צומת שערכו זהה לערכו של לפחות אחד מבניו ואותו בן הינו עץ צחי. כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר אמת אם העץ הינו עץ צחי. ושקר אם לא.

שאלה 35 Treepage18Targuil35

עץ בינארי יקרא עמית אם לכל צומת שהינו בן יחיד יש ערך זוגי. כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר אמת אם העץ הינו עץ עמית. ושקר אם לא.

שאלה 36

עץ בינארי יקרא מענין אם לכל צומת בעץ שיש נכד עש בן יחיד. כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר אמת אם העץ הינו עץ מענין. ושקר אם לא.

שאלה 37 Treepage19Targuil37

עץ בינארי יקרא ארזי אם כל צומת שהינו עלה הוא גם בו יחיד. כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תחזיר אמת אם העץ הינו עץ ארזי. ושקר אם לא.

תרגילים בשינוי עץ

שאלה 38

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תעדכן את ערכו של כל צומת שהינו עלה להיות כערכו של אביו.

שאלה 39

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תוסיף לכל בן יחיד אח שמכיל ערך הקטן ב1 מערכו שלו.

שאלה 40

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תמחק כל צומת שהינו בן יחיד ויש לו רק בן יחיד.

שאלה 41

כתוב ב-C# פעולה חיצונית המקבלת עץ בינארי T שהצמתים שלו הם מטיפוס שלם. הפעולה תוסיף לכל עלה בו בן שמאלי שערכו זהה לערכו של אב העלה.

תרגילים על 2 עצים.

שאלה 49

כתוב ב-C# פעולה חיצונית המקבלת 2 עצים בינאריים שהצמתים שלהם מטיפוס שלם. הפעולה תחזיר אמת אם 2 העצים זהים לחלוטין. ושקר אם לא.

שאלה 50

כתוב ב-C# פעולה חיצונית המקבלת 2 עצים בינאריים שהצמתים שלהם מטיפוס שלם. הפעולה תחזיר אמת אם 2 העצים הינם תמונת ראי (מבחינת המבנה). ושקר אם לא.

שאלה 51

כתוב ב-C# פעולה חיצונית המקבלת 2 עצים בינאריים שהצמתים שלהם מטיפוס שלם. הפעולה תחזיר אמת העץ השני הינו צאצא של העץ הראשון.

שאלה 52

כתוב ב-C# פעולה חיצונית המקבלת 2 עצים בינאריים שהצמתים שלהם מטיפוס שלם. הפעולה תחזיר את מספר הקשתות בין העץ הראשון לעץ השני. (הנח שהעץ השני הינו צאצא של העץ הראשון).

שאלה 53

כתוב ב-C# פעולה חיצונית המקבלת 3 עצים בינאריים שהצמתים שלהם מטיפוס שלם הנח שהעץ הראשון הינו אב קדמון של שני העצים האחרים. הפעולה תחזיר אמת אם 2 העצים נמצאים בגובה זהה. ושקר אם לא.

תרגילים ליצירת עצים

צייר כל אחד מהעצים הבאים:

```
//////This is the tree number 1
BinNode<int> b1=new BinNode<int>(1);
BinNode<int> b2 = new BinNode<int>(new BinNode <int>(2),3,new
BinNode <int>(8));
b1.SetLeft(new BinNode <int>(4));
b1.SetRight(b2);
VisualBinTree<int>.DrawTree(b1);

////////////////////////////////////////this is the tree number 2
BinNode<int> t1 = new BinNode<int>(8);
BinNode<int> t2 = new BinNode<int>(7);
BinNode<int> t3 = new BinNode<int>(11);
BinNode<int> t4 = new BinNode<int>(t1, 5, t2);
t2 = new BinNode<int>(22);
t2.SetRight(t3);
t4.GetRight().SetRight(t2);
VisualBinTree<int>.DrawTree(t4);

////////////////////////////////////////this is the tree number 3
BinNode<int> a = new BinNode<int>(8);
a.SetRight(new BinNode<int>(3));
a.GetRight().SetRight(new BinNode<int>(1));
VisualBinTree<int>.DrawTree(a);

////////////////////////////////////////this is the tree number 4
BinNode<int> c1 = new BinNode<int>(2);
BinNode<int> c2 = new BinNode<int>(8);
BinNode<int> c3 = new BinNode<int>(c1,2,null);
c2.SetRight(new BinNode<int> (9));
c3.SetRight(c2);
BinNode<int> c4 = new BinNode<int>(11);
c4.SetLeft(new BinNode<int>(12));
c3.SetLeft(c4);
VisualBinTree<int>.DrawTree(c3);

////////////////////////////////////////this is the tree number 5
BinNode<int> x = new BinNode<int>(7);
x.SetLeft(new BinNode<int>(1));
x.SetRight(new BinNode<int>(6));
x.GetLeft().SetRight(new BinNode<int>(4));
x.GetRight().SetRight(new BinNode<int>(2));
BinNode<int> b = new BinNode<int>(10);
b.SetLeft(new BinNode<int>(100));
b.SetRight(new BinNode<int>(200));
b.GetRight().SetRight(a);
VisualBinTree<int>.DrawTree(b);

////////////////////////////////////////this is the tree number 6
BinNode<string> y1 = new BinNode<string>("A");
BinNode<string> y2 = new BinNode<string>("B");
BinNode<string> y3 = new BinNode<string>("C");
y3.SetRight(new BinNode<string>("P"));
y2.SetLeft(new BinNode<string>("F"));
y1.SetRight(y2);
y3.GetRight().SetLeft(new BinNode<string>(y1, "T", y2));
y3.SetLeft(new BinNode<string>("Y"));
y3.GetLeft().SetRight (new BinNode<string>("G"));
VisualBinTree<string>.DrawTree(y3);
```

