

## חלק א – עלייכם לענות על כל השאלות בחלק זה במחברת הבחינה

### שאלה 1 (25 נקודות)

כתבו שיטה סטטית רקורסיבית בוליאנית המקבלת מערך של מספרים שלמים arr. השיטה צריכה להחזיר true אם אפשר לחלק את איברי המערך לשתי קבוצות שונות בגודלו (כלומר מספר האיברים בהם זהה) כך שסכום האיברים בשתי הקבוצות שווה. אם אי אפשר לעשות זאת, השיטה תחזיר false.

**חתימת השיטה היא:**

```
public static boolean equalSplit (int[] arr)
```

**דוגמאות:**

- עבור המערך הבא:

0	1	2	3	4	5
-3	5	12	14	-9	13

השיטה תחזיר true כי  $13 + -9 + 13 = 12 + -3 + 5 + 14$  וגם בכלל קבוצה יש 3 איברים

- עבור המערך הבא:

0	1	2	3	4	5
-3	5	-12	14	-9	13

השיטה תחזיר false כי אמם  $13 + -9 + 13 = -12 + 5 + 14 + -3$  אבל מספר האיברים בשתי הקבוצות שונה (באחת יש ארבעה איברים ובשנייה רק שניים).

- עבור המערך הבא:

0	1	2	3	4
-3	5	-12	14	-9

השיטה תחזיר false כי אין חלוקה של איברי המערך לשתי קבוצות שווות בגודלו.

**השיטה שתכתבו צריכה להיות רקורסיבית ללא שימוש בלולאות כלל. כך גם כל שיטות העזר שתכתבו (אם תכתבו) לא יכולות להכיל לולאות.**

אפשר להשתמש בהעמסת-יתר (overloading). אפשר להניח שהערך אינו null ואינו ריק.

אין לשנות את תוכן המערך (אפילו לא זמנית), ולא להשתמש בערך עז.

אין צורך לדאוג ליעילות השיטה! אבל כמובן צריך לשימוש לב לא לעשות קריאות רקורסיביות מיותרות!

**אל תשכחו לתעד את מה שתכתבם!**

לְאַבָּא לְכָהֵר תִּשְׁעִיר

## שאלה 2 (25 נקודות)

בשאלה זו נתיחס למערכות דו-ממדיים ריבועים, כלומר, מספר השורות והעמודות שווה (נניח שהוא שווה ל- $n$ ).

לצורך השאלה נניח כי הוא חזקה שלמה של 2.

עבור מערך זה, נגדיר חלוקה פנימית שלו לארבעה רובעים בגודל  $\frac{n}{2} \times \frac{n}{2}$ , ממוספרים מ-1 עד 4 באופן הבא:

1	2
4	3

נאמר שהמערך הוא **סיבובי** (circular) אם כל האיברים ברובע 1 קטנים ממש מכל אלו שברובע 2, וכל אלו שברובע 2 קטנים ממש מכל אלו שברובע 3, וכל אלו שברובע 3 קטנים ממש מכל אלו שברובע 4.

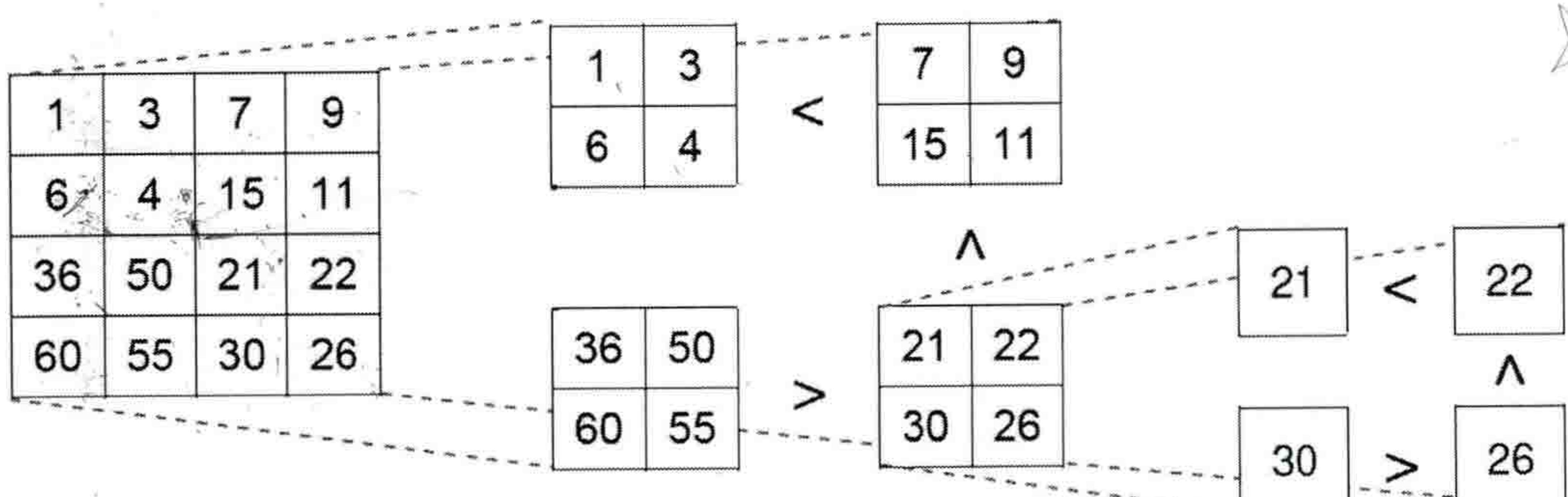
למשל, המערך הבא הוא סיבובי:

6	1
9	5
7	3

1	4
9	5

לשם הנוחות, נגדיר גם כל מערך בגודל  $1 \times 1$  כמערך סיבובי.

נאמר שמערך בגודל  $n \times n$  הוא **מומין-סיבובי** (circular sorted) אם הוא סיבובי, ארבעת הרובעים שלו סיבוביים, וכן הלאה עד לרובעים בגודל  $1 \times 1$ .  
לדוגמא, המערך הבא ממון-סיבובי:



עליכם לכתוב שיטה סטטית בוליאנית לחיפוש במערך ממון-סיבובי. השיטה מקבלת כפרמטרים את מערך דו-ממדי `mat` שהוא ממון-סיבובי, וערך לחיפוש `num`. אם הערך `num` נמצא במערך `mat`, השיטה תחזיר `true` ותדפיס את מספר השורה ומספר העמודה שבו נמצא המספר `num`. אם הערך `num` לא נמצא במערך `mat`, השיטה תחזיר `false` ולא יודפס כלום.

חתימת השיטה היא:

```
public static boolean search (int [][] mat, int num)
```

לען לכתיבת שולחין

**לדוגמא,**

אם המערך mat הוא המערך המצויר לעיל, והמספר num הוא 22, השיטה תחזיר את הערך true, ווינדפסו השורות הבאות:

row = 2

col = 3

אם המערך mat הוא המערך המצויר לעיל, והמספר num הוא 23, השיטה תחזיר את הערך false, ולא יונדפס כלום.

אתם יכולים להניח שהמערך mat אינו null והוא ממויין-סיבובית. איןכם צריכים לבדוק זאת.

**שימו לב:**

השיטה שתכתבו צריכה להיות עילה לכל הניתן, גם מבחינת סיבוכיות הזמן וגם מבחינת סיבוכיות המקום. תשובה שאינה עילה מספיק כלומר, שתהייה בסיבוכיות גדולה יותר מזו הנדרשת לפתרון הבעיה תקבל מעט נקודות בלבד.

**מה סיבוכיות זמן הריצה וסיבוכיות המקום של השיטה שתכתבם?  
הסבירו תשובהיכם.**

**אל תשחחו לטעד את מה שתכתבם!**

לְאַלְפָאַלְפִּיתְרָאַלְפִּים

**חלק ב - את התשובות לשאלות 3-5 יש לכתוב על גבי השאלה.  
לא נבדוק תשובות שייכתבו במקום אחר!**

**שאלה 3 (18 נקודות)**

נניח שהמחלקה Node שלהלו מ眞מת עצ' בינו'.

```
public class Node
{
    private int _number;
    private Node _leftSon, _rightSon;

    public Node (int number)
    {
        _number = number;
        _leftSon = null;
        _rightSon = null;
    }

    public int getNumber() {return _number; }
    public Node getLeftSon() {return _leftSon; }
    public Node getRightSon() {return _rightSon; }
}
```

המחלקה BinaryTree מאגדת בתוכה שיטות סטטיות לטיפול **בעץ בינו'.**  
בין השיטות נתונה השיטה printSecret הבאה, מקבלת שורש של עצ' בינו' ומספר שלם כລשחו.

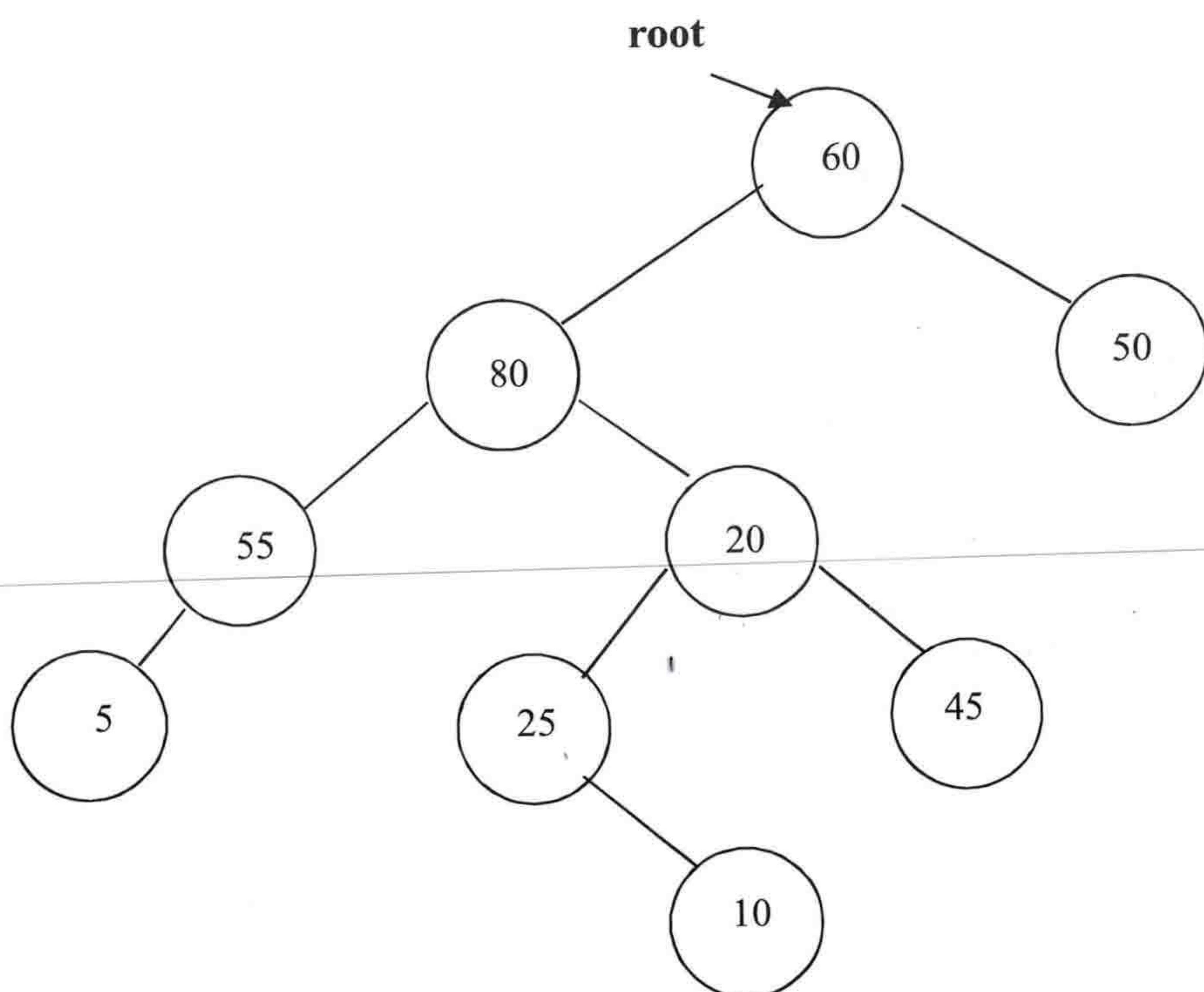
```
public static boolean printSecret(Node node, int target)
{
    if (node == null)
        return false;
    if (node.getNumber() == target)
        return true;

    if (printSecret(node.getLeftSon(), target)
    || printSecret(node.getRightSon(), target))
    {
        System.out.print(node.getNumber() + " ");
        return true;
    }

    return false;
}
```

לְאַמְתִּיבָה כְּסֻתְּרֵתֶךָ

נתון העץ הבינרי הבא, ששורשו הוא root :



ענו על הסעיפים הבאים :

(3 נק') א. בעקבות הקריאה `BinaryTree.printSecret(root, 40)`, מה יודפס ואייזה ערך תחזיר השיטה `printSecret` ?

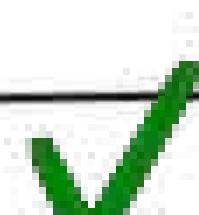
התשובה היא :

✓ false

הערך שיוחזר הוא :

יודפס (משמאל לימין) :

      60    45    25    10



(4 נק') ב. בעקבות הקריאה `BinaryTree.printSecret(root, 10)`, מה יודפס ואייזה ערך תחזיר השיטה `printSecret` ?

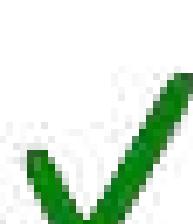
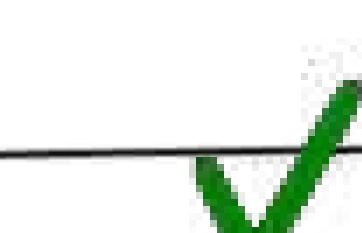
התשובה היא :

✓ true

הערך שיוחזר הוא :

יודפס (משמאל לימין) :

      25    20    80    60



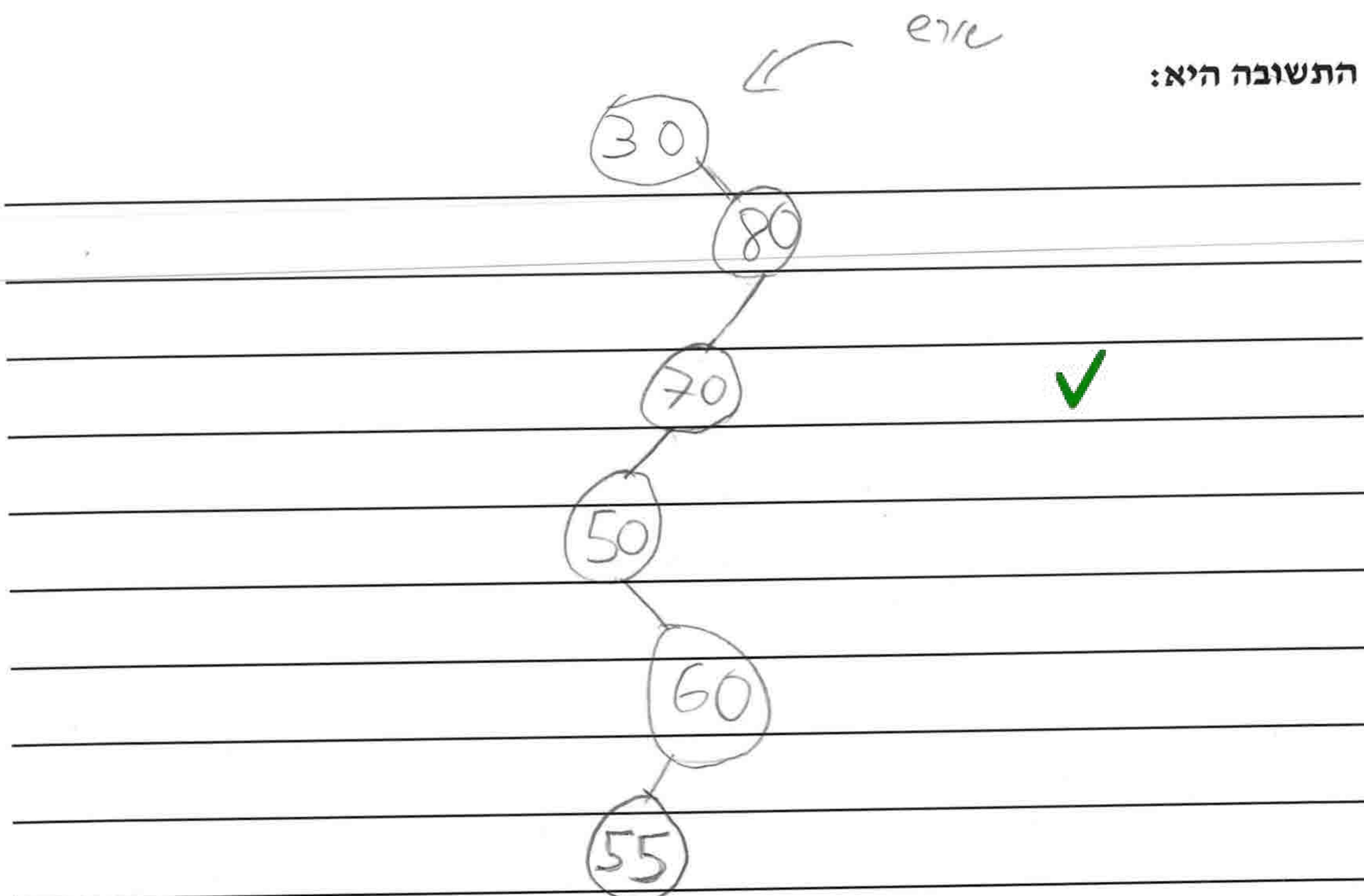
לְאַלְפָיִם תְּשׁוֹרֵת כְּסֻתָּרֶת



(5 נק') ג. עליכם לצייר עץ חיפוש בינרי (Binary Search Tree) שורשו root כך שאם נקרא לשיטה printSecret עם השורש root והמספר 55 יוחזר הערך true ויתודפס הפלט (משמאל לימין):

60 50 70 80 30

אם לא קיים עץ חיפוש ב内幕 כזה, עליהם להסביר מדוע.



(6 נק') ד. מה מבצעת השיטה `printSecret` באופן כללי כשהיא מקבלת כפרמטר שורש של עץ `binri root` ומספר `target`? שימו לב, עליהם לחת תיאור ממצאה של מה עשוה השיטה באופן כללי, ולא תיאור של מה עשוה כל שורה בשיטה, או איך היא מבצעת זאת. **כלומר, מה המשמעות של הערך שהשיטה מחזירה, ומה המשמעות של הדפסה? התיחסו למקרי קצה!**

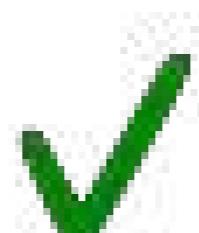
### התשובה היא:

לפעמים מושג קיומו של פיקט במאגרים נקרא **target - file** (הTARGET הנקרא **target-file** בפונקציית **getchar()**)

לדוגמה, אם נכתוב פונקציית **getchar()** כזו:

```
int getChar() {  
    char c = fgetc(stdin);  
    if (c == EOF) return -1;  
    else return c;  
}
```

ההפקה תאפשר לנו לשלוח פיקטים מהמסך למאגר, ולקבלם בחזרה.

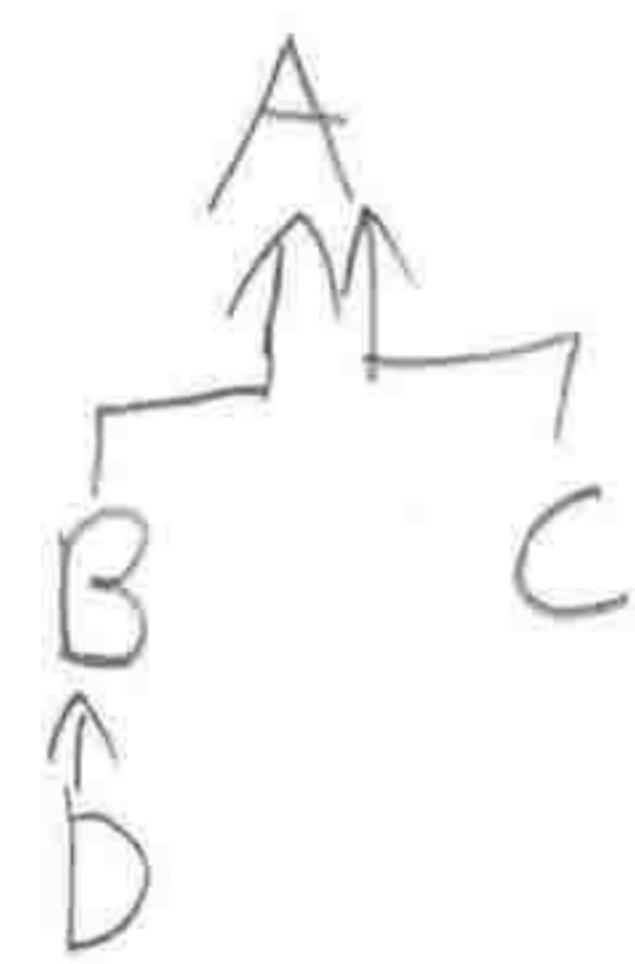


לְאַחֲרֵיכֶם תָּלִיכֶם

שאלה 4 (16 נקודות)

נתון פרויקט שהוגדרו בו המחלקות שאליהן. כל אחת בקובץ נפרד, כמוובן.

```
public class A {  
    protected int _x;  
  
    public boolean something(Object obj) {  
        System.out.println("1");  
        return super.equals(obj);  
    }  
}  
//-----//  
public class B extends A {  
  
    public boolean something(Object obj) {  
        System.out.println("2");  
        return super.something(obj);  
    }  
  
    public boolean something(B obj) {  
        System.out.println("3");  
        return super.something(obj);  
    }  
  
    public boolean something(A obj) {  
        System.out.println("4");  
        return super.something(obj);  
    }  
}  
//-----//  
public class C extends A {  
  
    public boolean something(B obj) {  
        System.out.println("5");  
        return super.something(obj);  
    }  
}
```



לְאַמְתִּיבָה וְכָלְבָד

```

public class D extends B {

    public boolean something(A obj) {
        System.out.println("6");
        return super.something(obj);
    }

    public boolean something(B obj) {
        System.out.println("7");
        return super.something(obj);
    }

    public boolean something(D obj) {
        System.out.println("8");
        return super.something(obj);
    }
}

```

```

public class Driver {

    public static void main(String[] args) {
        A a1 = new A();
        B b1 = new B();
        C c1 = new C();
        D d1 = new D();

        Object a2 = new A();
        A b2 = new B();
        A c2 = new C();
        D d2 = new D();
        A d3 = new D();

        // כל יופיעו הפקודות //
    }
}

```

לְאַתָּה לְכִינֵּת תְּמִימִינָה

להלן נתונות 8 פקודות. התייחסו אליהן כדיוון נמצאות בשיטה main וכתבו מה קורה לאחר כל פקודה.

אם הפקודה לא עוברת קומפילציה, כתבו למה. אם יש שגיאה בזמן ריצה, כתבו למה.  
אם הכל תקין, כתבו מה יהיה הפלט שיודפס בעקבות הפקודה.

אין קשר בין הפקודות!

1. b1.something(a2);

2  
1  
✓

2. c1.something(d2);

5  
1  
✓

3. d1.something(b1);

7  
3  
1  
✓

4. a2.something(c1);

Object נשלח - נזק נזק Object  
Object something נשלח נזק נזק Object  
Object נשלח נזק נזק Object  
✓

5. b2.something(d3);

2  
1  
✓

6. c2.something(c2);

1  
✓

7. d2.something(a1);

6  
4  
1  
✓

8. d3.something(c2);

2  
1  
✓



לְאַמְתָּה אֲסֵרֶת תִּלְעִיבָּה

**שאלה 5 (16 נקודות)**

נתונה המחלקה `IntNode` הבאה, המייצגת איבר בראשימה:

```
public class IntNode {
    private int _value;
    private IntNode _next;

    public IntNode(int val, IntNode n) {
        _value = val;
        _next = n;
    }

    public int getValue() {
        return _value;
    }
    public IntNode getNext() {
        return _next;
    }

    public void setValue(int v) {
        _value = v;
    }
    public void setNext(IntNode node) {
        _next = node;
    }
}
```

נתונה רשימה מקוישרת של מספרים שלמים, הממומשת בעזרת המחלקה `IntList` שלහן:

```
public class IntList
{
    private IntNode _head;
    public IntList( ) {
        _head = null;
    }
    public IntList(IntNode h ) {
        _head = h;
    }
    // כאן יש עוד בנאים ושיתות... //
    // המשך המחלקה בעמוד הבא //
}
```

לְאַתָּה לְכָה תִּמְלִיכָה

```

public int f()
{
    IntNode temp = _head;
    int c = 0;
    while (temp != null)
    {
        c++;
        temp = temp.getNext();
    }
    return c;
}

public int secret2021B62(int x)
{
    int curr = 0, result = f() + 1; 22
    int temp1 = 0, temp2 = 0;
    IntNode ptr1 = _head, ptr2 = _head;
    while (ptr2 != null)
    {
        while (curr <= x && ptr2 != null)
        {
            curr += ptr2.getValue(); 23
            ptr2 = ptr2.getNext(); 24
            temp2++; 25
        }

        while (curr > x && ptr1 != null)
        {

            if (temp2 - temp1 < result)
                result = temp2 - temp1; 43

            curr -= ptr1.getValue(); 23
            ptr1 = ptr1.getNext();
            temp1++;
        }
    }
    return result;
}

. . . // other methods
}

```

**אתם יכולים להניח שהרשימה מלאה במספרים שלמים או שליליים בלבד!**

לְאַתָּה נִסְעֵת תְּמִימִינָה

בטענות להלן, נסמן את איברי הרשימה כמספרים מופרדים בפסיקים, בתוך סוגרים מסולסלים.  
כך לדוגמה, נסמן  $\{ -4, 9, 12, 21 \}$  את הרשימה null  $\rightarrow -4 \rightarrow 9 \rightarrow 12 \rightarrow 21 \rightarrow$

**סעיף א (1 נקודות)**

מה הערך אותו תחזיר השיטה f כSEQUENCE איתה על הרשימה  $\{ 3, 6, 2, 7 \}$ ?

התשובה היא:



4

**סעיף ב (2 נקודות)**

מה מבצעת השיטה f באופן כללי? הסבירו בקצרה מה השיטה עושה ולא **כיצד** היא מבצעת זאת.  
שים לב, עלייכם לתת תיאור ממזכה של מה עושה השיטה **באופן כללי**, ולא תיאור של מה עושה כל שורה בשיטה, או איך היא מבצעת זאת. כמובן, עלייכם לכתוב מה המשמעות של הערך שמוחזר מהשיטה f, כשהיא מופעלת על רשימה כלשהי.

התשובה היא:

השיטה f מקבלת כArgument רשימה כלשהי ומחזירה סדרה של זוגים (x,y) WHERE x=y

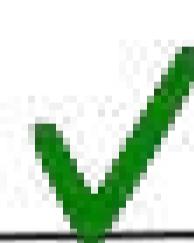
השיטה f מקבלת כArgument רשימה כלשהי ומחזירה סדרה של זוגים (x,y) WHERE x=y

**לגבי סעיפים ג – ה להלן נתונה הרשימה**  $\{ 1, 4, 13, 6, 0, 19 \}$ . list = { 1 , 4 , 13 , 6 , 0 , 19 }

**סעיף ג (2 נקודות)**

איזה ערך תחזיר הקראיה לשיטה list.secret2021B62 (22)?

התשובה היא:

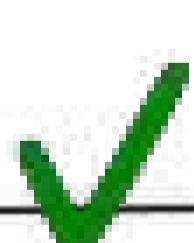


3

**סעיף ד (2 נקודות)**

איזה ערך תחזיר הקראיה לשיטה list.secret2021B62 (43)?

התשובה היא:



7

(המשך השאלה בעמוד הבא)

לְאַבָּא לְכָהֵר תִּשְׁלֵחַ



לען לכתור רשות