

מס' שאלון - 470

3

במרץ 2020

מס' מועד 87

סמסטר 2020א

20441 / 4

שאלון בחינת גמר

20441 - מבוא למדעי המחשב ושפת Java

משך בחינה: 3 שעות

בשאלון זה 13 עמודים

מבנה הבחינה:

- קראו בעיון את ההנחיות שלהלן:
- * בבחינה יש חמש שאלות.
 - * כל התכניות צריכות להיות מתועדות היטב.
 - יש לכתוב תחילה בקצרה את האלגוריתם וכל הסבר נוסף הדרוש להבנת התכנית.
 - יש לבחור בשמות משמעותיים למשתנים, לפונקציות ולקבועים שבתכנית.
 - תכנית שלא תתועד כנדרש לעיל תקבל לכל היותר 85 % מהניקוד.
 - * יש להקפיד לכתוב את התכניות בצורה מבנית ויעילה.
 - תכנית לא יעילה לא תקבל את מלוא הנקודות.
 - * אם ברצונכם להשתמש בתשובתכם בשיטה או במחלקה הכתובה בחוברת השקפים, אין צורך שתעתיקו את השיטה או את המחלקה למחברת הבחינה. מספיק להפנות למקום הנכון, ובלבד שההפניה תהיה מדויקת (פרמטרים, מיקום וכו').
 - * אין להשתמש במחלקות קיימות ב-Java, חוץ מאלו המפורטות בשאלות הבחינה.
 - * יש לשמור על סדר; תכנית הכתובה בצורה בלתי מסודרת עלולה לגרוע מהציון.
 - * בכתיבת התכניות יש להשתמש אך ורק במרכיבי השפה שנלמדו בקורס זה
 - אין להשתמש במשתנים גלובליים!
 - * את התשובות לשאלות 3-5 יש לכתוב על גבי השאלון. לא נבדוק תשובות שייכתבו במקום אחר!
 - * אפשר לתעד בעברית. אין צורך בתיעוד API.

חומר עזר:

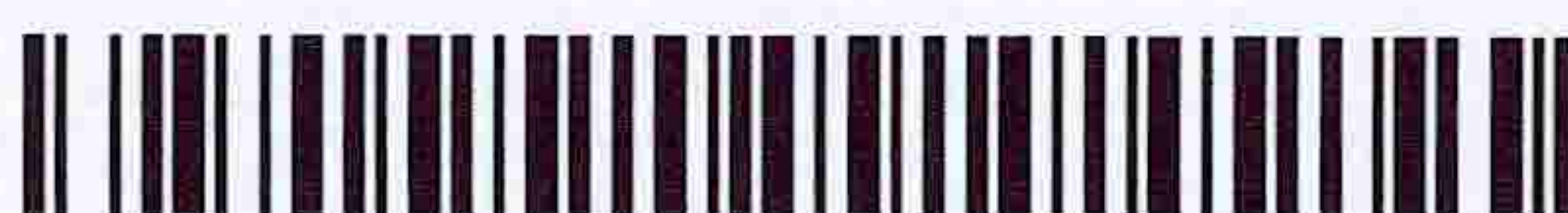
ספר הלימוד : java software solutions מאת : lewis/loftus
חוברת השקפים של הקורס של ד"ר אמיר גורן ותמר וילנר.
יחידות 1-6, 7-12. מותרות הערות בכתב יד, ע"ג הספרים.
אין להכניס חומר מודפס או כל חומר אחר מכל סוג שהוא.

החזירו

למשגיח את השאלון

וכל עזר אחר שקיבלתם בתוך מחברת התשובות

בהצלחה !!!



חלק א – עליכם לענות על כל השאלות בחלק זה במחברת הבחינה

שאלה 1 (25 נקודות)

נתון מערך דו-ממדי mat , בגודל $n \times m$ (n שורות ו- m עמודות) שערכיו הם מספרים שלמים.

- נגדיר **מסלול** ($path$) במערך אם הוא מקיים את התנאים הבאים:
 1. המסלול מתחיל בתא הראשון במערך - שורה ראשונה ועמודה ראשונה, ומסתיים בתא האחרון במערך - שורה אחרונה ועמודה אחרונה.
 2. אפשר ללכת רק בכיוונים למטה וימינה. כלומר, מתא $mat[i][j]$ אפשר ללכת רק לתא $mat[i+1][j]$ (למטה) או לתא $mat[i][j+1]$ (ימינה).
- נגדיר **פניה** ($turn$) במסלול אם הולכים ימינה ומיד אחר כך למטה, או למטה ומיד אחר כך ימינה.

לדוגמא, במערך mat להלן מסומנות בחיצים שתי פניות:

 - מהתא $(1,0)$ לתא $(2,0)$ ומשם לתא $(2,1)$
 - מהתא $(0,2)$ לתא $(0,3)$ ומשם לתא $(1,3)$

	0	1	2	3	4
0					
1					
2					

עליכם לכתוב שיטה סטטית רקורסיבית המקבלת מערך דו-ממדי, ומספר שלם k כלשהו. השיטה צריכה להחזיר את מספר המסלולים השונים בהם יש בדיוק k פניות.

לדוגמא,

במטריצה מסדר 3×3 הבאה, כתבנו בתוך התא את האינדקסים שלו (ממילא הערכים שנמצאים בתאים לא רלוונטיים לשאלה זו):

(0,0)	(0,1)	(0,2)
(1,0)	(1,1)	(1,2)
(2,0)	(2,1)	(2,2)

- אם $k = 1$ אז השיטה צריכה להחזיר את הערך 2 שכן יש שני מסלולים במטריצה שמתחילים בתא הראשון ומסתיימים בתא האחרון שכל אחד מהם מכיל פניה אחת בלבד. והם:

$(0,0) \rightarrow (0,1) \rightarrow (0,2) \rightarrow (1,2) \rightarrow (2,2)$

$(0,0) \rightarrow (1,0) \rightarrow (2,0) \rightarrow (2,1) \rightarrow (2,2)$

- אם $k = 2$ אז השיטה צריכה להחזיר את הערך 2 שכן יש שני מסלולים במטריצה שמתחילים בתא הראשון ומסתיימים בתא האחרון שכל אחד מהם מכיל שתי פניות. והם:

$(0,0) \rightarrow (0,1) \rightarrow (1,1) \rightarrow (2,1) \rightarrow (2,2)$

$(0,0) \rightarrow (1,0) \rightarrow (1,1) \rightarrow (1,2) \rightarrow (2,2)$

- אם $k = 3$ אז השיטה צריכה להחזיר את הערך 2 שכן יש שני מסלולים במטריצה שמתחילים בתא הראשון ומסתיימים בתא האחרון שכל אחד מהם מכיל שלוש פניות. והם:

$(0,0) \rightarrow (0,1) \rightarrow (1,1) \rightarrow (1,2) \rightarrow (2,2)$

$(0,0) \rightarrow (1,0) \rightarrow (1,1) \rightarrow (2,1) \rightarrow (2,2)$

- אם $k = 4$ אז השיטה צריכה להחזיר את הערך 0 שכן אין אף מסלול במטריצה שמכיל ארבע פניות.

חתימת השיטה היא:

```
public static int totalWays(int[][] mat, int k)
```

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

אפשר להשתמש בהעמסת-יתר (overloading).

מותר לשנות את המערך במהלך השיטה, אבל בסופה הוא צריך לחזור למצבו המקורי.

אין צורך לדאוג ליעילות השיטה, אבל כמובן שצריך לשים לב לא לעשות קריאות רקורסיביות מיותרות! אל תשכחו לתעד את מה שכתבתם!


```

public static int totalWays(int [][] mat,int k ) {
    return totalWays(mat,k,0,0,0);
}

private static int totalWays(int [][] mat,int k ,int i,int j,int lastI) {
    if(i < 0 || j < 0 || i >= mat.length || j >= mat[0].length || k < 0)
        return 0;
    if (i == mat.length-1 && j == mat.length-1 && k == 0)
        return 1;

    if(i == 0 && j == 0)

        return totalWays(mat,k,i+1,j,i);
    if(lastI == i) {
        return totalWays(mat,k,i+1,j,i) + totalWays(mat,k,i,j+1,i);
    }else {
        return totalWays(mat,k,i+1,j,i) + totalWays(mat,k-1,i,j+1,i);
    }
}
}

```


שאלה 2 (25 נקודות)

נתון מערך חד-ממדי a המלא במספרים שלמים חיוביים הממוינים בסדר עולה ממש (כלומר, אין מספרים שווים).

כתבו שיטה סטטית המקבלת את המערך הממוין a ומספר שלם נוסף num . השיטה צריכה להדפיס את כל שלשות המספרים שבמערך a שהמכפלה שלהם היא num .

לדוגמא:

- אם המערך a הוא:

$$a = \{1, 2, 3, 4, 5, 7, 8, 9\}$$

והערך $num = 40$

אזי השלשות שיודפסו יהיו

1	5	8
2	4	5

$$1 * 5 * 8 = 2 * 4 * 5 = 40 \quad \text{שכן}$$

- אם המערך a הוא:

$$a = \{1, 3, 6, 8\}$$

והערך $num = 40$

אז לא יודפס כלום

שכן, אין אף שלשה במערך שהמכפלה שלה היא 40.

חתימת השיטה היא:

```
public static void printTriplets (int[] a, int num)
```

שימו לב:

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

מה סיבוכיות זמן הריצה וסיבוכיות המקום של השיטה שכתבתם?
הסבירו תשובתכם.

אל תשכחו לתעד את מה שכתבתם!

המשך הבחינה בעמוד הבא


```
public static void printTriplets(int [] a, int num) {  
    int low;  
    int high;  
    for(int i=0;i <= a.length-2;i++) {  
        low = i+1;  
        high = a.length-1;  
        while(low < high){  
            if(a[i] * a[low] * a[high] == num) {  
                System.out.println(a[i] + " " + a[low] + " " + a[high]);  
            }  
            if(a[i] * a[low] * a[high] < num) {  
                low++;  
            }else {  
                high--;  
            }  
        }  
    }  
}
```


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

שאלה 3 (17 נקודות)

נניח שהמחלקה Node שלהלן מממשת עץ בינרי.

```
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 בינרי.

בין השיטות נתונות השיטות what ו-secret הבאות:

```
public static int what (Node t)
{
    if (t == null)
        return 0;
    return 1 + Math.max (what (t.getLeftSon()),
                        what (t.getRightSon()));
}
```



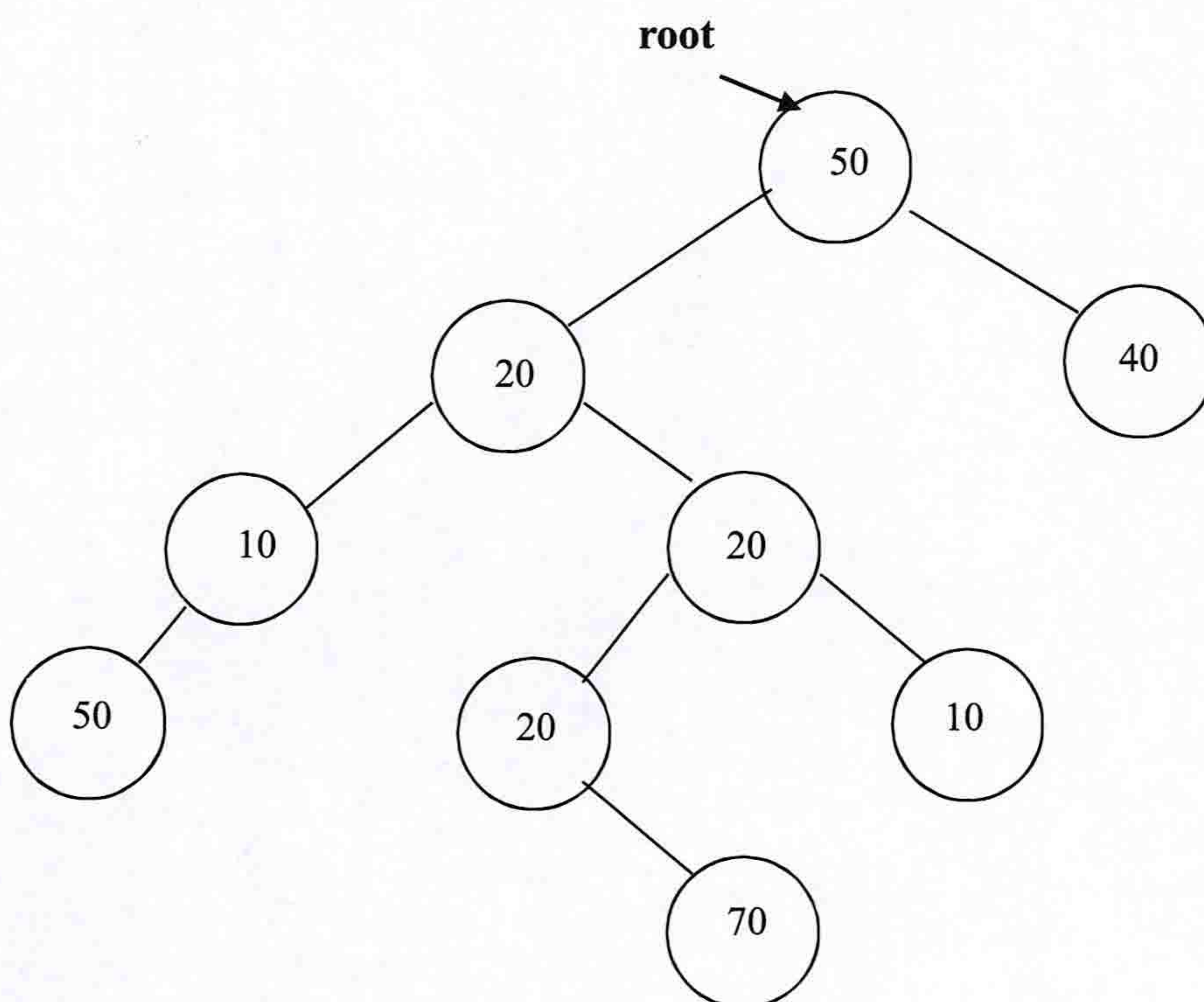
```
public static void secret(Node node, int[] a)
{
    secret (node, 0, a);
}

private static void secret (Node node, int k, int[] a)
{
    if (node == null)
        return;
    a[k] += node.getNumber();
    secret (node.getLeftSon(), k + 1, a);
    secret (node.getRightSon(), k + 1, a);
}
```

נניח שקיימת במחלקה BinaryTree גם השיטה printArr הבאה, המדפיסה את הערכים שבתאי מערך חד-ממדי הניתן לה כפרמטר:

```
public static void printArr(int arr[], int n)
{
    for (int i = 0; i < n; i++)
        System.out.print(arr[i]+ " " );
    System.out.println();
}
```

נתון העץ הבינרי הבא, ששורשו הוא root



ענו על הסעיפים הבאים:

(2 נק') (א) איזה ערך תחזיר השיטה `what` בעקבות הקריאה `BinaryTree.what(root)`?

התשובה היא:

5

(3 נק') (ב) מה מבצעת השיטה `what` באופן כללי כשהיא מקבלת כפרמטר שורש של עץ בינרי `root`?

שימו לב, עליכם לתת תיאור ממצה של מה עושה השיטה באופן כללי, ולא תיאור של

מה עושה כל שורה בשיטה, או איך היא מבצעת זאת. כלומר, מה המשמעות של הערך

שהשיטה מחזירה?

התשובה היא:

השיטה בודקת את עומק העץ מהשורה עד לעלה כולל השורש במידה וריק מחזיר `null`

(5 נק') (ג) מה יודפס כתוצאה מביצוע קטע הקוד הבא, כאשר `root` הוא שורש העץ לעיל:

```
int num = what(root);
int [] a = new int[num];
secret(root, a);
printArr(a, num);
```

התשובה היא:

50 60 30 80 70

(7 נק') (ד) מה מבצעת השיטה `secret` באופן כללי כשהיא מקבלת כפרמטר שורש של עץ בינרי `root`,

ומערך `a`? שימו לב, עליכם לתת תיאור ממצה של מה עושה השיטה באופן כללי, ולא

תיאור של מה עושה כל שורה בשיטה, או איך היא מבצעת זאת. כתבו מה המשמעות של

הערכים שמודפסים לאחר הפקודה `printArr(a, num)`

התשובה היא:

השיטה מכניסה לכל איבר במקום איקס המערך את סכום ערכי הצמתים בעץ שרמתם היא איקס לאחר הפקודה מודפסים סכום ערכי הצמתים ברמה 0 ואז ברמה 1 וכן הלאה

נתון פרויקט שהוגדרו בו המחלקות A, B שלהלן. כל אחת בקובץ נפרד, כמונן.

```
public class A
{
    private int _val;

    public A() {
        _val = 1;
    }

    public A(int val) {
        _val = val;
    }

    public int getVal() {
        return _val;
    }

    public boolean equals(Object other) {
        System.out.println("AObject ");
        if (other instanceof A)
            return (_val == ((A) other)._val);
        return false;
    }
}

//-----

public class B extends A
{
    private String _st;

    public B() {
        _st = "B";
    }

    public B(String st, int val) {
        super(val);
        _st = st;
    }

    public String getSt() {
        return _st;
    }

    public boolean equals(Object other) {
        System.out.println("BObject ");
        if (other instanceof B)
            return (_st.equals(((B) other)._st) &&
                    (getVal() == ((B) other).getVal()));
        return false;
    }
}
```



```
public boolean equals(A other) {
    System.out.println("BA ");
    if (other instanceof B)
        return (_st.equals(((B) other)._st) &&
            (getVal() == ((B) other).getVal()));
    return false;
}

public boolean equals(B other) {
    System.out.println("BB ");
    return (_st.equals(other._st) &&
        (getVal() == other.getVal()));
}
}
```

בפרויקט נמצאת גם המחלקה Driver ובה השיטה main שלהלן:

```
public static void main (String[] args)
{
    A a1 = new A();
    A a2 = new A(5);
    A ab = new B();
    B b1 = new B("B",1);
    B b2 = new B("B",5);

    A c = (A)b2;

    // *** //
}
```

סעיף א (5 נקודות):

לאחר הרצת הקוד לעיל בשיטה main שבמחלקה Driver, המצהיר על האובייקטים שבתכנית, מלאו את הטבלה להלן כך שהיא תכיל בכל תא את הערך של התכונה המתאימה של האובייקט. אם לאובייקט אין תכונה כזו, סמנו ב- X

האובייקט	_val	_st
a1	1	X
a2	5	X
ab	1	B
b1	1	B
b2	5	B
c	5	X



סעיף ב (10 נקודות):

עבור כל אחת מן ההוראות 1-10 בצעו:

- כתבו את ההוראה במקום `/**` בשיטה `main` של המחלקה `Driver`
- כתבו מה יהיה הפלט בעקבות הוספה של כל אחת מההוראות שלהלן: שימו לב לכתוב את תשובותיכם אך ורק על השורה המסומנת. **כתבו את הפלט של השורה הנוספת בלבד, אין לכתוב את הפלט של השורות הקודמות לה.**
- אין קשר בין שורות הקוד. בכל פעם יש להניח ש-6 שורות הקוד שכתובות בשיטה `main` כבר קיימות והשורה הנבדקת היא השורה השביעית.

1. `if (a1.equals(b1)) System.out.println(1);`

 **AObject 1**

2. `if (b1.equals(a1)) System.out.println(2);`

 **BA**

3. `if (a1.equals(ab)) System.out.println(3);`

 **AObject 3**

4. `if (ab.equals(a1)) System.out.println(4);`

 **BObject**

5. `if (b1.equals(ab)) System.out.println(5);`

 **AObject 5**

6. `if (ab.equals(b1)) System.out.println(6);`

 **BObject 6**

7. `if (a1.equals(a2)) System.out.println(7);`

 **AObject**

8. `if (b1.equals(b2)) System.out.println(8);`

 **BB**

9. `if (a1.equals(c)) System.out.println(9);`

 **AObject**

10. `if (c.equals(a1)) System.out.println(10);`

 **Bobject**



שאלה 5 (18 נקודות)

נתונה המחלקה הבאה, המייצגת איבר ברשימה:

```

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 מקושרת של מספרים שלמים, הממומשת בעזרת המחלקה IntList שלהלן:

```

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

    // המשך המחלקה בעמוד הבא
    // כאן יש עוד בנאים ושיטות...
}

```



```

public int something()
{
    int m = 0;

    IntNode ptr = _head;
    while (ptr.getNext() != null)
    {
        int c = 1;
        while ((ptr.getNext() != null) &&
            (ptr.getValue() < ptr.getNext().getValue()))
        {
            ptr = ptr.getNext();
            c++;
        }

        while ((ptr.getNext() != null) &&
            (ptr.getValue() > ptr.getNext().getValue()))
        {
            ptr = ptr.getNext();
            c++;
        }

        if (c > m)
            m = c;
    }

    return m;
}

. . . // other methods
}

```

אתם יכולים להניח שהרשימה מלאה במספרים שלמים.

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

סעיף א (2 נקודות)

נתונה הרשימה $list = \{ 2, 6, 3, 4, 7, 4, 6, 1, 3 \}$. איזה ערך תחזיר הקריאה לשיטה `list.f()` ?
 התשובה היא

9

סעיף ב (3 נקודות)

מה מבצעת השיטה `f` באופן כללי כשהיא מופעלת על רשימה כלשהי? הסבירו בקצרה מה השיטה עושה ולא כיצד היא מבצעת זאת.
 שימו לב, עליכם לתת תיאור ממצה של מה עושה השיטה באופן כללי, ולא תיאור של מה עושה כל שורה בשיטה, או איך היא מבצעת זאת. כלומר, עליכם לכתוב מה המשמעות של הערך שמוחזר מהשיטה `f`, כשהיא מופעלת על רשימה כלשהי. התייחסו למקרי קצה.

התשובה היא

השיטה סוכמת את איברי הרשימה, במידה והרשימה ריקה
 נחזיר 0.

סעיף ג (3 נקודות)

נתונה הרשימה $list = \{ 2, 6, 3, 4, 7, 4, 6, 1, 3 \}$. איזה ערך תחזיר הקריאה לשיטה `list.something()` ?
 התשובה היא

4

סעיף ד (3 נקודות)

נתונה הרשימה $list = \{ 6, 7, 1, 2, 3, 4, 5 \}$. איזה ערך תחזיר הקריאה לשיטה `list.something()` ?
 התשובה היא

5

סעיף ה (7 נקודות)

מה מבצעת השיטה `something` באופן כללי כשהיא מופעלת על רשימה כלשהי? הסבירו בקצרה מה השיטה עושה ולא כיצד היא מבצעת זאת. אפשר להניח שאין שני מספרים זהים צמודים ברשימה.

שימו לב, עליכם לתת תיאור ממצה של מה עושה השיטה באופן כללי, ולא תיאור של מה עושה כל שורה בשיטה, או איך היא מבצעת זאת. כלומר, עליכם לכתוב מה המשמעות של הערך שמוחזר מהשיטה `something`, כשהיא מופעלת על רשימה כלשהי. התייחסו למקרי קצה.

התשובה היא

השיטה בודקת כמה איברים עולים או יורדים ממש ברצף יש ברשימה עד ל-X מסויים ומחזירה את כמות האיברים הגדולה יותר של העולים או היורדים ממש ברצף.
 במידה ואין, נחזיר 0.

בהצלחה :