

"ח באלו תשפ"א

מוס' שאלון - 477

26

באוגוסט 2021

92 מוס' מועד

סמסטר 2021ב

20441 / 4

### שאלון בחינת גמר

## Java - מבוא למדעי המחשב ושפת Java

משך בחינה: 3 שעות

בשאלון זה 15 עמודים

### מבנה הבחינה:

קראו בעיון את ההנחיות שלhallon:

\* בבחינה יש חמיש שאלות.

\* כל התכניות צריכות להיות מתועדות היטב.

יש לכתוב תחילת בקצרה את האלגוריתם וכל הסבר נוסף הדרוש להבנת התכנית.

יש לבחור בשמות משתנים למשתנים, לפונקציות ולקבועים שבתכנית.

תכנית שלא תتوודع כנדרש לעיל תקבל לכל היותר % 85 מהניקוד.

\* יש להקפיד לכתוב את התכניות בצורה מבנית ויעילה.

תכנית לא יעילה לא תקבל את מלא הנקודות.

\* אם ברצונכם להשתמש בתשובותם בשיטה או בחלוקת הכתיבה בחוברת השקפים,

אין צורך שתעתיקו את השיטה או את המחלוקת מהבחינה. מספיק להפנות

למקום הנכון, וב惟ב ששהפניה תהיה מדויקת (פרמטרים, מיקום וכו').

\* אין להשתמש בחלוקת קיימות ב- Java , חוץ מآل המפורטות בשאלות הבחינה.

\* יש לשמר על סדר; תכנית הכתובה בצורה בלתי מסודרת עלולה לגרוע מהציוון.

\* בכתב התכניות יש להשתמש אך ורך במרכיבי השפה שנלמדו בקורס זה

אין להשתמש בשמות גLOBליים!

\* אפשר לטעד בעברית. אין צורך בתיעוד API .

את התשובות לשאלות 5-3 יש לכתוב על גבי השאלון בלבד.

לא יבדקו תשובות שייכתו במקום אחר.

### חומר עזר:

חוברות השקפים 6-1, 12-7.

אין להכין חומר מודפס נוסף או חומר אחר מכל סוג.


בהצלחה !!!

### החזירו

למשגיח את השאלון

וכל עזר אחר שקיבלתם בתור מחברת התשובות





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

### שאלה 1 (25 נקודות)

נתון מערך חד-ממדי `arr` שערכוו הם מספרים שלמים חיוביים ממש (אין צורך לבדוק זאת). בהינתן מספר שלם `num` כלשהו, אנו רוצים לבדוק אם אפשר להגיע למספר `num` על ידי חיבור וחיסור חלק מהמספרים במערך `arr`, ובכמה אופנים.

לדוגמא,

אם המערך `arr` הוא להלן והמספר `num` הוא 4

|   |   |   |   |
|---|---|---|---|
| 0 | 1 | 2 | 3 |
| 1 | 3 | 6 | 2 |

אפשר להגיע למספר 4 כך:

- $+3 +1 = 4$
- $+6 -3 +1 = 4$
- $+2 +3 -1 = 4$
- $+2 +6 -3 -1 = 4$
- $-2 +6 = 4$

כלומר, יש לנו 5 ביטויים חישbowים עם פעולות חיבור וחיסור בלבד, שמערבים את איברי המערך `arr` ותוצאותיהם הם המספר  $4 = num$ .

שים לב שאנו סופרים פעם אחת כל ביטוי, ולא סופרים את הביטויים שימושם בחוק החלוף. כלומר, אם ספכנו את הביטוי  $1 + 3 +$  לא נספר את הביטוי  $3 + 1 +$ .

כתבו שיטה סטטית וקורסיבית שמקבלת כפרמטרים מערך `arr` של שלמים חיוביים ממש, ומספר שלם `num`, ומוחירה את מספר הביטויים החישbowים שימושם בפעולות חיבור וחיסור על איברי המערך (לא בהכרח כולם), ומגיעים לתוצאה `num`.

כמו כן, השיטה צריכה להדפיס את הביטויים אלו. **ההדפסה תכלול את המספר השלם והסימן פלוס (+) או מינוס (-) שלפניו, ורואה לאחריו.**

ראו את הדוגמא להלן.

תא כהנומינאלית

**דוגמא:**

אם נפעיל את השיטה על המערך arr לעיל ועל המספר 3 = nun יודפסו הביטויים הבאים :

+2 +1

-2 +6 -1

+3

+6 -3

והשיטה תחזיר את הערך 4.

**חתימת השיטה היא:**

```
public static int printExpr (int[] arr)
```

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

אפשר להשתמש בהעמסת-יתר (overloading).

שימוש לב, שיטה שלא תבצע את ההפסה של הביטויים החבוניים, אלא רק תחזיר (באופן נכון) את מספרם, קיבל לכל יותר 15 נקודות.

אין צורך לדאוג ליעילות השיטה, אבל כמובן צריך לשימוש לב לא לעשות קריאות רקורסיביות מיותרות! אל תשכח לטע את מה שתכתבם!

לען לכתיב רשלין

## שאלה 2 (25 נקודות)

נתו מערך חד-ממדית שמכיל רק אפסים (0) ואחדים (1).

כתבו שיטה סטטית מקבלת כפרמטרים מערך  $a$  כזה ומספר שלם אי-שלילי `longestSequence`. השיטה צריכה לחזיר את האורכה ביותר של הערך  $a$ , שכוללת  $k$ . השיטה צריכה להחזיר את האורכה הרציפה של הערך  $a$ , לכל היתר  $k$  אפסים.

לדוגמא:

אם נתנו המערך  $a$  הבא:

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|----|----|
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0  | 0  |

- השיטה תחזיר 4 הסדרה באורך 4 המתחילה ממוקם 6 היא הארוכה ביותר עם  $k=0$  אפסים עבור  $k=0$
  - השיטה תחזיר 7 (הסדרה באורך 7 המתחילה ממוקם 3 היא הארוכה ביותר עם  $k=1$  אפסים עבור  $k=1$ )
  - השיטה תחזיר 10 (הסדרה באורך 10 המתחילה ממוקם 0 מכילה בדיק  $k=2$  אפסים עבור  $k=2$ )
  - השיטה תחזיר 11 (הסדרה באורך 11 המתחילה ממוקם 0 מכילה מספר אפסים מסוימי האפשרי  $3$  אפסים) עבור  $k=3$
  - השיטה תחזיר 12 (הסדרה באורך 12 המתחילה ממוקם 0 מכילה מספר אפסים מסוימי האפשרי  $4$  אפסים, והיא בעצם המערך כולל)

#### **חתימת השיטה:**

```
public static int longestSequence (int[] a, int k)
```

שםו לב:

**גדולה יותר מזו הנדרשת לפתרונו הביעיה תקבל מעט נקודות בלבד.**  
מבחן סיבוכיות המקום. תשובה שאינה עילה מספיק כלל, שתהייה בסיבות  
השיטה שתתבאו צריכה להיות כל הניתן, גם מבחן סיבוכיות הזמן וגם

**הסבירו תשובה לכם. אל תטע את מה שתכתבם!**

לְאַלְפָיִם תְּשׁוֹלֵם

**חלק ב - את התשובות לשאלות 3- 5 יש לכתוב על גבי השאלה!  
לא נבדוק תשבות שיכתבו במקום אחר!**

**שאלה 3 (18 נקודות)**

נניח שהמחלקה Node שלhallן מימושה עצם ביניי.

```
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 **BinarySearchTree** מאגדת בתוכה שיטות סטטיות לטיפול בעץ חיפוש ביניי.

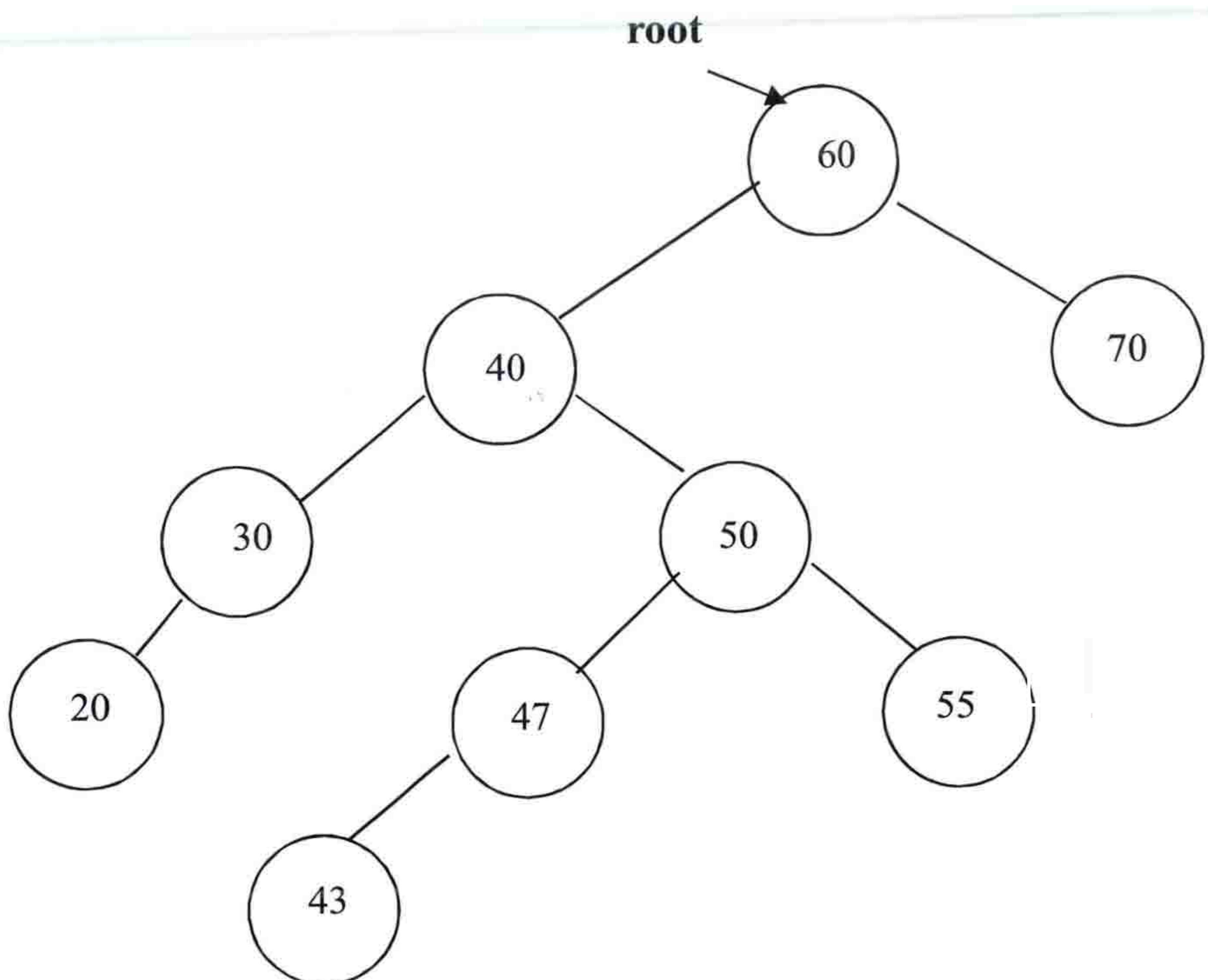
נתונה השיטה **find** הבאה, המתקבלת שורש של עצם חיפוש ביניי ומספר שלם כלשהו.

```
public static int find(Node root, int num)
{
    if (root == null)
        return -1;
    if (root.getNumber() == num)
        return num;
    else if (root.getNumber() < num) {
        int k = find(root.getRightSon(), num);
        if (k == -1)
            return root.getNumber();
        else
            return k;
    }
    else if (root.getNumber() > num)
        return find(root.getLeftSon(), num);
    return -1;
}
```

לְאַלְפָיִם תְּשׁוֹלֵחַ

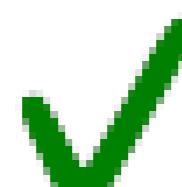
נתון עץ חיפוש הבינרי הבא, ששורשו הוא root.

אנחנו מניחים שכל הערכים שנמצאים בעץ הם חיוביים!



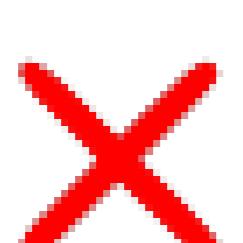
ענו על הסעיפים הבאים:

(2 נק') (א) איזה ערך תחזיר השיטה **find** בעקבות הקריאה `?BinarySearchTree.find(root, 43)`



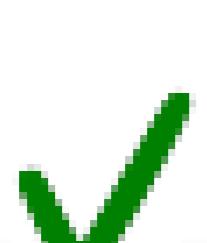
התשובה

(2 נק') (ב) איזה ערך תחזיר השיטה **find** בעקבות הקריאה `?BinarySearchTree.find(root, 59)`



התשובה היא:

(2 נק') (ג) איזה ערך תחזיר השיטה **find** בעקבות הקריאה `?BinarySearchTree.find(root, 15)`



התשובה היא:

לען לערוך טענות

(3 נק') (ד) מה מבצעת השיטה **find** באופן כללי כשהיא מקבלת כפרמטר שורש של עץ חיפוש בינרי root ומספר שלם num כלשהו? שימו לב, עליכם לתת תיאור ממצה 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? התיחסו למקרי קצה!**

**התשובה היא:**

---



---

#### (6 נק') (ח) הסעיף הזה לא קשור לסעיפים הקודמים!

עליכם לציר עץ חיפוש בינרי (Binary Search Tree) שבו שורשו root כך שם נדפיס אותו **בסדר סופי** (post order) יתקבל הפלט הבא (משמאל לימין):

30 20 50 55 60 60 75 40 70 85

אם לא קיים עץ חיפוש בינרי כזה, עליכם להסביר מדוע.

**להזכירכם**, השיטה **postOrder** המדפיסה עץ בינרי בסדר סופי ממומשת כך:

```
public static void printPostOrder (Node root)
{
    if (root != null)
    {
        printPostOrder(root.getLeftSon());
        printPostOrder(root.getRightSon());
        System.out.print (root.getNumber() + " ");
    }
}
```

**התשובה היא (ציירו את העץ במסגרת להלן):**

6  
(3)

לְאַלְפָיִם שָׁמֶן

שאלה 4 (16 נקודות)

נתון פרויקט שהוגדרו בו המחלקות A,B,C ו-Driver שלහן. כל אחת בקובץ נפרד, כמו הבן.

```
public abstract class A
{
    protected int _x;

    public A() {
        _x = 1;
    }

    public abstract int foo(int x);

    public void foo(A a) {
        _x = a._x;
    }
}
```

```
public class B extends A
{
    public B() {
        super();
    }

    public B(int val) {
        _x = val;
    }

    public B(A val) {
        _x = foo(val._x);
    }

    public int foo(int x) {
        return _x + x;
    }

    public void foo(B b) {
        _x = _x * b._x;
    }
}
```

לְאַמְתִּיבָה וְלְאַמְתִּיבָה

```

public class C extends B {

    public C() {
        super();
    }

    public C(int val) {
        super(val);
    }

    public int foo(int x) {
        return _x - x;
    }

    public void foo(A a) {
        if (a instanceof C)
            _x = _x - a._x;
        else
            super.foo(a);
    }

    public void foo(B b) {
        super.foo(b);
        _x++;
    }

    public void foo(C c) {
        _x = c._x - 1;
    }
}

```

**במחלקה Driver נמצאת השיטה main שלහלן:**

```

public static void main(String[] args)
{
    A a;
    B b;
    C c;

    - כאן יופיעו הפקודות שבסעיפים הבאים -
}

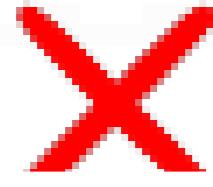
```

עבור כל סעיף להלן, הקיפו בעיגול את התשובה המתאימה ביותר. עלייכם להחליט האם קטע הקוד גורם לשגיאת קומpileציה, שגיאת ריצה, או תקין. במידה והקוד תקין השלימו את התשובה וכתבו מה יהיה ערכו של `_x`.

**אין קשר בין הסעיפים!**

לען לא ככיתר רשותיהם

/\*1\*/ a = new A();

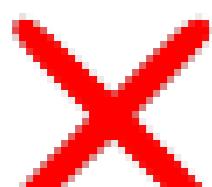
 a.foo(2);

טע הקוד גורם לשגיאת קומPILEציה והיא

טע הקוד גורם לשגיאת ריצה והיא

טע הקוד תקין וערך התכונה x באובייקט a הוא:

/\*2\*/ a = new B();

 b = new C();

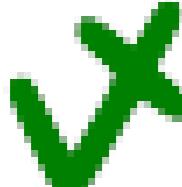
b.foo(a);

טע הקוד גורם לשגיאת קומPILEציה והיא

טע הקוד גורם לשגיאת ריצה והיא \_

טע הקוד תקין וערך התכונה x באובייקט a הוא:

/\*3\*/ a = new C();

 b = new C();

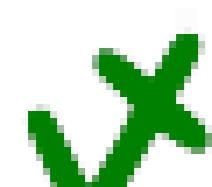
((C)b).foo(a);

קטע הקוד גורם לשגיאת קומPILEציה והיא

קטע הקוד גורם לשגיאת ריצה והיא

קטע הקוד תקין וערך התכונה x באובייקט a

/\*4\*/ C c1 = new C();

 b = new C();

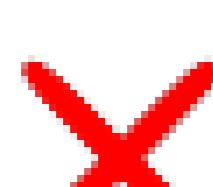
b.foo(c1);

קטע הקוד גורם לשגיאת קומPILEציה והיא

קטע הקוד גורם לשגיאת ריצה והיא

קטע הקוד תקין וערך התכונה x באובייקט

/\*5\*/ c = new C();

 c.foo(c);

טע הקוד גורם לשגיאת קומPILEציה והיא

טע הקוד גורם לשגיאת ריצה והיא \_

טע הקוד תקין וערך התכונה x באובייקט c הוא:

לְאַמְתִּיבָה וְלְאַמְתִּיבָה

/\*6\*/ C c1 = new C(2);  
✓  
 a = new C(3);  
 c1.foo((B)a);

הקוד גורם לשגיאת קומPILEציה והיא \_\_\_\_\_

הקוד גורם לשגיאת ריצה והיא \_\_\_\_\_

הקוד תקין וערך התכונה x \_ באובייקט א הוא :

/\*7\*/ b = new B(2);  
✗  
 Object o = b;  
 o.foo(4);

הקוד גורם לשגיאת קומPILEציה והיא \_\_\_\_\_

הקוד גורם לשגיאת ריצה והיא \_\_\_\_\_

הקוד תקין וערך התכונה x \_ באובייקט ב הוא :

/\*8\*/ C c1 = new C(2);  
 a = new C(3);  
 c1.foo(a);

✗ זקד גורם לשגיאת קומPILEציה והיא \_\_\_\_\_  
 זקד גורם לשגיאת ריצה והיא \_\_\_\_\_  
 זקד תקין וערך התכונה x \_ באובייקט א הוא :

3  
 (4)

לְאַמְתִּיבָה כְּכֹלֶת תְּשִׁיבָה

**שאלה 5 (16 נקודות)**

נתונה המחלקה IntNode 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 מקוורת של מספרים שלמים, המוממשת בעזרת המחלקה IntList שללן:

```
public class IntList  
{  
    private IntNode _head;  
    public IntList( ) {  
        _head = null;  
    }  
  
    public boolean what1 (IntList list)  
    {  
        IntNode h1 = _head;  
        IntNode h2 = list._head;  
        while ((h1 != null) && (h2 != null))  
        {  
            if (h1.getValue() != h2.getValue())  
                return false;  
            h1 = h1.getNext();  
            h2 = h2.getNext();  
        }  
        return true;  
    }  
}
```

לען לכתיבת תשלילים

```

public boolean what2 (IntList list)
{
    IntNode h1 = _head;

    while (h1 != null)
    {
        boolean found = false;
        IntNode h2 = list._head;
        while ((h2 != null) && (!found))
        {
            if (h1.getValue() == h2.getValue())
                found = true;
            h2 = h2.getNext();
        }
        if (!found)
            return false;
        h1 = h1.getNext();
    }
    return true;
}
} //end of class IntList

```

**אתם יכולים להניח שהרשימה מלאה במספרים שלמים.**

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

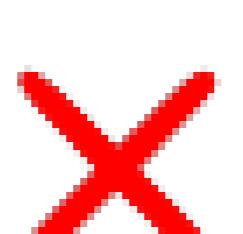
**ענו על ששת הסעיפים להלן:**

**בכל הסעיפים נתונה הרשימה `{1, 2, 3, 4, 5}`**

#### **סעיף א (2 נקודות)**

תנו דוגמא של רשימה בת 4 איברים לפחות בשם list כך שהקריאה `testList.what1(list)` תחזיר `true` והקריאה `testList.what2(list)` תחזיר `false`.  
אם לא יכולה להיות רשימה כזו, כתבו זאת וסבירו מדוע.

**התשובה היא:**



לְאַמְתִּיבָה כְּבָשָׂר וְלְבָשָׂר

**סעיף ב (2 נקודות)**

תנו דוגמא של רשימה בת 4 איברים לפחות בשם list כך שהקריאה (list.what1(list)) תחזיר true והקריאה (list.what2(list)) תחזיר false.  
אם לא יכולה להיות רשימה כזו, כתבו זאת וסבירו מדוע.  
**התשובה היא:**

**X**

---

---

---

---

**סעיף ג (2 נקודות)**

תנו דוגמא של רשימה בת 4 איברים לפחות בשם list כך שהקריאה (list.what1(list)) תחזיר false והקריאה (list.what2(list)) תחזיר true.  
אם לא יכולה להיות רשימה כזו, כתבו זאת וסבירו מדוע.  
**התשובה היא:**

**X**

---

---

---

---

**סעיף ד (2 נקודות)**

תנו דוגמא של רשימה בת 4 איברים לפחות בשם list כך שהקריאה (list.what1(list)) תחזיר false והקריאה (list.what2(list)) תחזיר true.  
אם לא יכולה להיות רשימה כזו, כתבו זאת וסבירו מדוע.  
**התשובה היא:**

**X**

---

---

---

---

לְאַמְתִּיבָה וְלְאַמְתִּיבָה

#### סעיף ה (4 נקודות)

מה מבצעת השיטה 1 what באופן כללי? הסבירו בקצרה מה השיטה עושה ולא **באיזה** היא מבצעת זאת.

שיםו לב, עלייכם לתת תיאור ממצה של מה עושה השיטה באופן כללי, ולא תיאור של מה עושה כל שורה בשיטה, או איך היא מבצעת זאת. כלומר, עלייכם לכתוב מה המשמעות של הערך שמוחזר מהשיטה 1 what, כשהיא מופעלת על רשימה כלשהי list1 ומקבלת כפרמטר רשימה אחרת list2.

**התשובה היא:**

---

---

---

---

#### סעיף ו (4 נקודות)

מה מבצעת השיטה 2 what באופן כללי? הסבירו בקצרה מה השיטה עושה ולא **באיזה** היא מבצעת זאת.

שיםו לב, עלייכם לתת תיאור ממצה של מה עושה השיטה באופן כללי, ולא תיאור של מה עושה כל שורה בשיטה, או איך היא מבצעת זאת. כלומר, עלייכם לכתוב מה המשמעות של הערך שמוחזר מהשיטה 2 what, כשהיא מופעלת על רשימה כלשהי list1 ומקבלת כפרמטר רשימה אחרת list2.

**התשובה היא:**

---

---

---

---

---

---

---

---

---

**בצלחה!**

לְאַבָּא כָּסֵף טָהוֹר