

כ"ג בשבט תשע"ה

מס' שאלון - 460

12

בפברואר 2015

סמסטר 2015א

20441 / 4

מס' מועד 83

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 שש 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.

\* בכתובת התכניות יש להשתמש אך ורק במרכיבי השפה שנלמדו בקורס זה

אין להשתמש במשתנים גלובליים!

\* את התשובות לשאלות 3 - 6 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.  
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**

עבור מערך דו-ממדי של מספרים שלמים, נגדיר **מסלול** (path) באורך k במערך להיות סדרה של k תאים, שכל אחד מהם סמוך לקודמו, כלומר, הוא אחד מארבעת שכניו במערך (שכן מעליו, מתחתיו, מימינו ומשמאלו, אך לא באלכסון לו), כך שהערכים בתאים מהווים סדרה עולה ממש של מספרים שלמים.

כתבו שיטה סטטית רקורסיבית המקבלת מערך דו-ממדי המכיל מספרים שלמים חיוביים בלבד (אין צורך לבדוק את תקינות תוכן המערך), ומחזירה את אורכו של המסלול הקצר ביותר המתחיל בתא הראשון (שמאלי עליון – שורה 0 ועמודה 0) ומסתיים בתא האחרון (ימני תחתון – שורה אחרונה ועמודה אחרונה).

לדוגמא, במערך שלהלן:

|   | 0  | 1  | 2  | 3  | 4   |
|---|----|----|----|----|-----|
| 0 | 3  | 13 | 15 | 28 | 30  |
| 1 | 40 | 51 | 52 | 29 | 30  |
| 2 | 28 | 10 | 53 | 54 | 53  |
| 3 | 53 | 12 | 55 | 53 | 60  |
| 4 | 70 | 62 | 56 | 20 | 80  |
| 5 | 80 | 81 | 90 | 95 | 100 |

ישנם לפחות 3 מסלולים, הראשון והאחרון באורך 10 והשני באורך 14:

- $(0,0) \rightarrow (0,1) \rightarrow (0,2) \rightarrow (0,3) \rightarrow (1,3) \rightarrow (1,4) \rightarrow (2,4) \rightarrow (3,4) \rightarrow (4,4) \rightarrow (5,4)$
- $(0,0) \rightarrow (0,1) \rightarrow (0,2) \rightarrow (1,2) \rightarrow (2,2) \rightarrow (3,2) \rightarrow (4,2) \rightarrow (4,1) \rightarrow (4,0) \rightarrow (5,0) \rightarrow (5,1) \rightarrow (5,2) \rightarrow (5,3) \rightarrow (5,4)$
- $(0,0) \rightarrow (0,1) \rightarrow (1,1) \rightarrow (1,2) \rightarrow (2,2) \rightarrow (3,2) \rightarrow (4,2) \rightarrow (5,2) \rightarrow (5,3) \rightarrow (5,4)$

ולכן השיטה תחזיר את הערך 10.

**חתימת השיטה:**

```
public static int shortestPath (int [][] mat)
```

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 חובה להחזיר אותו למצבו ההתחלתי בסופה.

אל תשכחו לתעד את מה שכתבתם!



## שאלה 2 - 25 נקודות

### סעיף א – 7 נקודות

כתבו שיטה סטטית בוליאנית **יעילה** המקבלת כפרמטרים מערך חד-ממדי arr המכיל מספרים שלמים, ומספר שלם נוסף num. **המערך ממזין בסדר עולה ממש.**

השיטה צריכה להחזיר true אם קיימים במערך arr **שני איברים שונים** שסכומם הוא num, ו-false אחרת.

**לדוגמא,** עבור המערך  $arr = \{-2, 3, 5, 7, 12\}$

אם הערך  $num = 17$  השיטה תחזיר true כיוון ש-  $5+12 = 17$

אם הערך  $num = 16$  השיטה תחזיר false כיוון שאין אף זוג איברים במערך arr שסכומם הוא 16.

### חתימת השיטה:

```
public static boolean sum2 (int [] arr, int num)
```

### סעיף ב – 18 נקודות

כתבו שיטה סטטית בוליאנית **יעילה** המקבלת כפרמטרים מערך חד-ממדי arr המכיל מספרים שלמים, ומספר שלם נוסף num. **המערך ממזין בסדר עולה ממש.**

השיטה צריכה להחזיר true אם קיימים במערך arr **שלושה איברים שונים** שסכומם הוא num, ו-false אחרת.

**לדוגמא,** עבור המערך  $arr = \{-2, 3, 5, 7, 12\}$

אם הערך  $num = 17$  השיטה תחזיר true כיוון ש-  $-2+7+12 = 17$

אם הערך  $num = 16$  השיטה תחזיר false כיוון שאין אף שלשה של איברים במערך arr שסכומם הוא 16.

### חתימת השיטה:

```
public static boolean sum3 (int [] arr, int num)
```

### שימו לב:

- **שתי השיטות שתכתבו צריכה להיות יעילות 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.
- אפשר להשתמש בשיטה מסעיף א בפתרון של סעיף ב (בזהירות...). גם מי שלא פתר את סעיף א, יכול להניח שיש פתרון ולהשתמש בו בסעיף ב.
- כתבו מה סיבוכיות הזמן ומה סיבוכיות המקום של כל שיטה שכתבתם.
- אל תשכחו לתעד את מה שכתבתם!



**חלק ב - את התשובות לשאלות 3-6 יש לכתוב על גבי השאלון.  
לא נבדוק תשובות שייכתבו במקום אחר!**

**שאלה 3 - 12 נקודות**

נניח שהמחלקה Node שלהלן מממשת עץ בינרי.

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

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

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

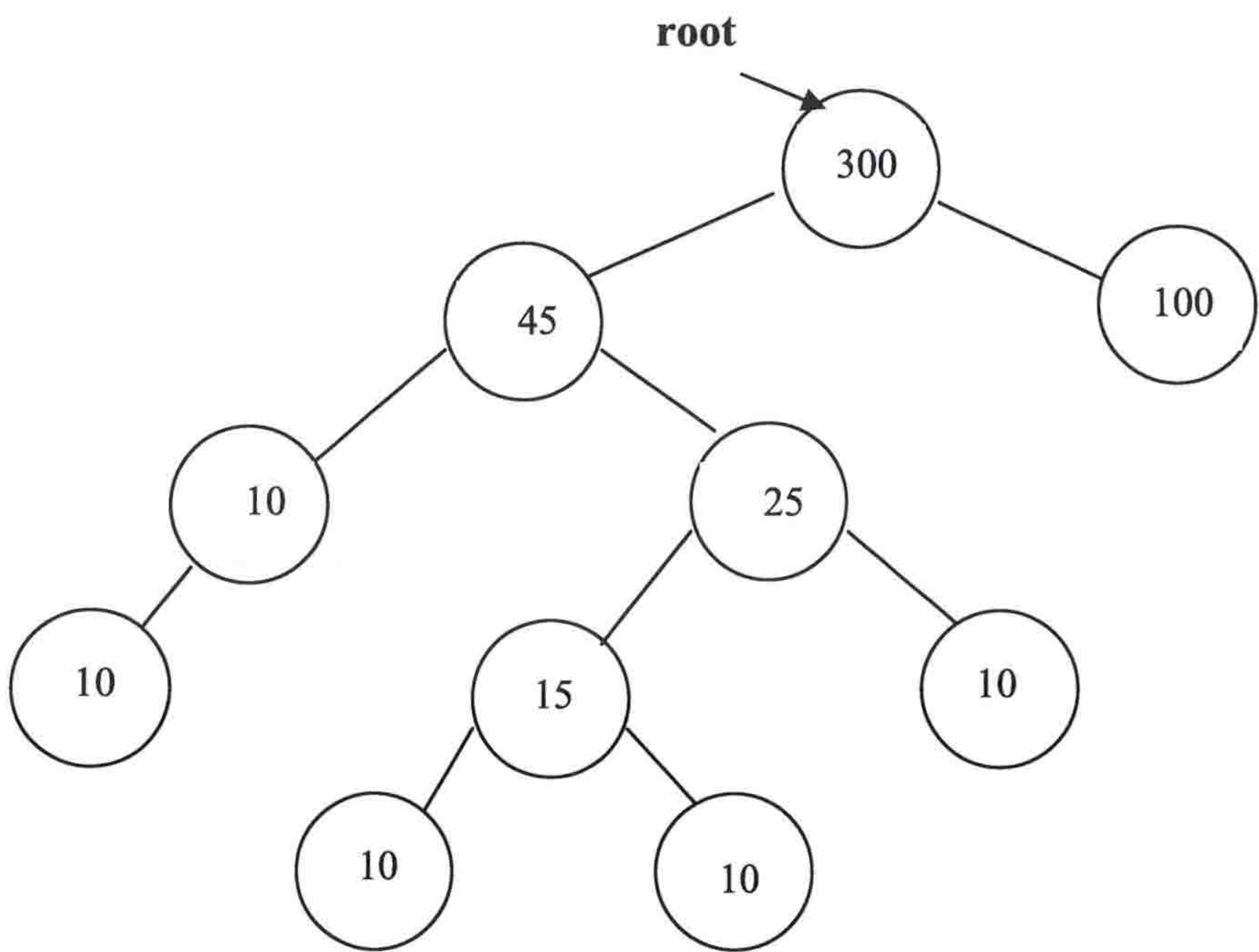
המחלקה BinaryTree מאגדת בתוכה שיטות סטטיות לטיפול בעץ בינרי.  
בין השיטות נתונות השיטות הבאות:

```
public static void what(Node t)
{
    what(t, "");
}

private static void what(Node t, String s)
{
    if (t != null)
        s = s + " " + t.getNumber();
    if ((t.getLeftSon()==null) && (t.getRightSon()==null))
        System.out.println(s);
    else
    {
        if (t.getLeftSon() != null)
            what (t.getLeftSon(), s);
        if (t.getRightSon() != null)
            what (t.getRightSon(), s);
    }
}
```



נתון העץ הבינרי הבא, ששורשו הוא root



ענו על הסעיפים הבאים:

5 נק' (i) איך ייראה הפלט בעקבות הקריאה `?BinaryTree.what (root)`  
התשובה היא: (לא בהכרח כל השורות יתמלאו)

---

---

---

---

---

---

7 נק' (ii) מה מבצעת השיטה `what` באופן כללי? 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.

התשובה היא:

---

---

---

---



נתונות המחלקות A, B, C, Driver הבאות (באותה חבילה):

```
public class A
{
    protected int _num;
    public A(int n)
    {
        _num = n;
    }
}

//-----//

public class B extends A
{
    public B(int n)
    {
        super(n);
    }

    public boolean equals(A a)
    {
        return _num == a._num;
    }
}

//-----//

public class C extends A
{
    public C(int n)
    {
        super(n);
    }

    public boolean equals(A a)
    {
        return a instanceof C && _num == a._num;
    }
}
```



```

public class Driver
{
    public static void main(String[] args)
    {
        A x1 = new A(10);
        A y1 = new B(10);
        B y2 = new B(10);
        A z1 = new C(10);
        C z2 = new C(10);

        // שורות הקוד יתווספו פה

    }
}

```

נניח שאנחנו כותבים שורות קוד שונות בשיטה main של המחלקה Driver במקום ההערה:  
 שורות הקוד יתווספו פה // בכל סעיף כתבו מה יהיה הפלט ונמקו תשובתכם  
**בקצרה. כמו כן ציינו איזו שיטה equals הופעלה (מאיזו מחלקה).**  
**סימון ללא נימוק לא יזכה את הכותב בנקודות!**

שימו לב – אין קשר בין שורות הקוד. בכל פעם אתם צריכים להניח שחמש שורות הקוד  
 שכתובות כבר במחלקה קיימות, והשורה הנבדקת היא השורה השישית ב-main.

1. System.out.println(y1.equals(y2));  
 יודפס true / false כי \_\_\_\_\_

השיטה equals שהופעלה נמצאת במחלקה B / C / Object

2. System.out.println(y2.equals(y1));  
 יודפס true / false כי \_\_\_\_\_

השיטה equals שהופעלה נמצאת במחלקה B / C / OBJ

3. System.out.println(z1.equals(z2));  
 יודפס true / false \_\_\_\_\_

השיטה equals שהופעלה נמצאת במחלקה B / C / OBJ



4. `System.out.println(z2.equals(z1));`

יודפס true / false כי \_\_\_\_\_

השיטה equals שהופעלה נמצאת במחלקה B / C / OBJ

5. `System.out.println(y2.equals(z2));`

יודפס true / false כי \_\_\_\_\_

השיטה equals שהופעלה נמצאת במחלקה B / C / OBJ

6. `System.out.println(z2.equals(y1));`

יודפס true / false כי \_\_\_\_\_

השיטה equals שהופעלה נמצאת במחלקה B / C / OBJ

7. `System.out.println(z2.equals(x1));`

יודפס true / false כי \_\_\_\_\_

השיטה equals שהופעלה נמצאת במחלקה B / C / OBJ

8. `System.out.println(x1.equals(z2));`

יודפס true / false כי \_\_\_\_\_

השיטה equals שהופעלה נמצאת במחלקה B / C / OBJ

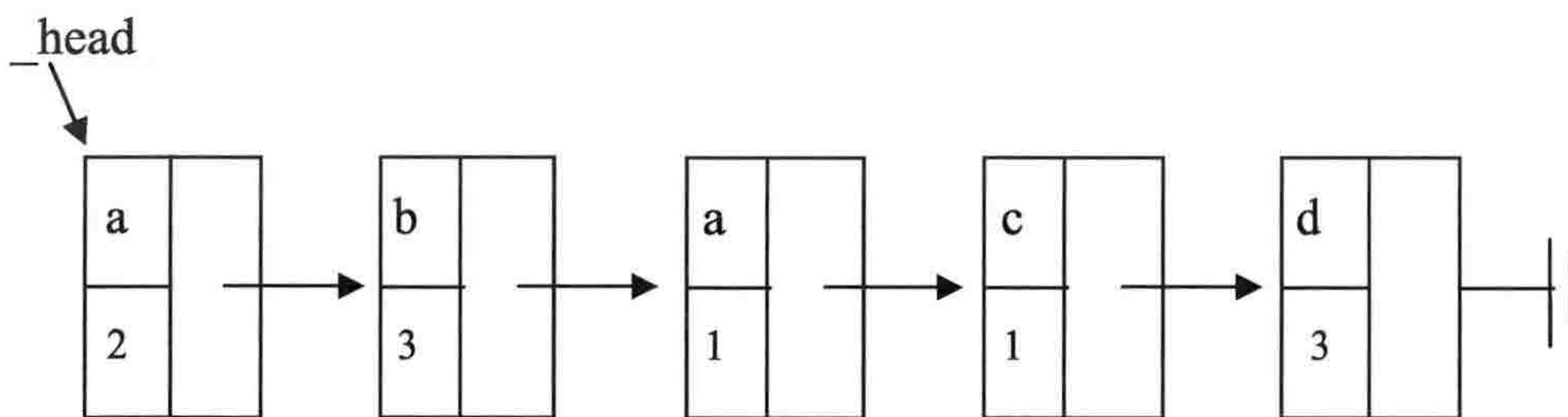


## שאלה 5 - 14 נקודות

נייצג מחרוזת תווים ברשימה מקושרת באופן הבא.  
בכל צומת ברשימה יהיו שלושה שדות:

1. האות עצמה;
2. מספר הפעמים שהיא מופיעה ברצף;
3. מצביע לצומת הבא ברשימה.

כך למשל המחרוזת "aabbbaacddd" תיוצג על-ידי הרשימה הבאה:



המחלקה CharNode שלהלן מייצגת צומת ברשימה:

```

public class CharNode
{
    private char _data;
    private int _value;
    private CharNode _next;

    public CharNode(char c, int val, CharNode n) {
        _data = c;
        _value = val;
        _next = n;
    }

    public CharNode getNext( )           { return _next; }
    public void setNext(CharNode node) { _next = node; }
    public int getValue()                 { return _value; }
    public void setValue(int v)           { _value = v; }
    public char getData()                 { return _data; }
    public void setData(char c)           { _data = c; }
}

```



```

public class StringList
{
    private CharNode _head;

    public StringList(String s) {...}

    public int what (char ch, int fr)
    {
        CharNode pos = _head;
        int c=0, c1=0;
        while (pos != null && c < fr)
        {
            c = c + pos.getValue();
            if (c <= fr)
                pos=pos.getNext();
        }
        if( pos == null) return -1;

        c1= help(ch,pos);
        if( c1==-1)
            return -1;
        if (c1 > 0)
            c1 = c1 - (c - fr);

        return fr + c1;
    }

    private int help( char ch, CharNode pos)
    {
        boolean found = false;
        int c = 0;
        while (pos != null && !found)
            if(pos.getData() == ch)
                found= true;
            else
            {
                c = c + pos.getValue();
                pos=pos.getNext();
            }
        if(found) return c;
        return -1;
    }
}

```

במחלקה StringList נתון בנאי המקבל כפרמטר מחרוזת תווים s ויוצר ממנה רשימה מקושרת המייצגת את המחרוזת s.



כמו כן נתונה שיטה what המקבלת כפרמטרים תו ch ומספר שלם fr ומחזירה מספר שלם.

### סעיף א (8 נקודות)

לכל טענה סמנו אם היא נכונה או לא נכונה.

א) אם נפעיל את השיטה what על הרשימה המייצגת את המחרוזת "abbabc", והפרמטרים יהיו  $ch1 = 'b'$  ו-  $fr = 2$ , השיטה תחזיר את הערך 3. ☐ נכון ☐ לא / ☐ נכון

ב) אם נפעיל את השיטה what על הרשימה המייצגת את המחרוזת "abbabc", פעם אחת עם הפרמטרים  $ch1 = 'a'$  ו-  $fr = 2$ , ופעם שניה עם הפרמטרים  $ch1 = 'a'$  ו-  $fr = 1$ , השיטה תחזיר אותו ערך. ☐ נכון ☐ לא / ☐ נכון

ג) אם נפעיל את השיטה what על הרשימה המייצגת את המחרוזת "abbccabc", והפרמטרים יהיו  $ch1 = 'c'$  ו-  $fr = 2$ , השיטה תחזיר את הערך 3. ☐ נכון ☐ לא / ☐ נכון

ד) אם נפעיל את השיטה what על הרשימה המייצגת את המחרוזת "abbccabc", והפרמטרים יהיו  $ch1 = 'c'$  ו-  $fr = 8$ , השיטה תעוף על שגיאת ריצה. ☐ נכון ☐ לא / ☐ נכון

### סעיף ב (6 נקודות)

מה מבצעת השיטה what באופן כללי? 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 what, כשהיא מופעלת על רשימה המייצגת מחרוזת כלשהי ומקבלת כפרמטרים תו ch1 ומספר שלם fr כלשהו. שימו לב למקרי הקצה.

אתם צריכים להתייחס למחרוזת המיוצגת ברשימה עליה מופעלת השיטה. (דוגמא לתשובה בכיוון זה היא: "השיטה מחזירה את מספר המופעים של ch1 במחרוזת")

---

---

---

---



במחלקה Q6 נתונות השיטות הסטטיות הבאות:

```
public class Q6
{
    public static int what(int []a)
    {
        int m = 0;
        for (int i=0; i<a.length; i++)
        {
            for (int j=a.length-1; j>-1; j--)
            {
                if (a[i] == a[j])
                {
                    int t = f(a, i, j);
                    if (t > m)
                        m = t;
                }
            }
        }
        return m;
    }

    private static int f(int []a, int i, int j)
    {
        int c = 0;
        while (i<a.length && j>-1)
        {
            if (a[i] ==a[j])
            {
                c++;
                i++;
                j--;
            }
            else
                return c;
        }
        return c;
    }
}
```



(2 נק') (i) אם נתון המערך a הבא:

|   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 |
| 7 | 1 | 2 | 9 | 7 | 2 | 1 |

איזה ערך השיטה תחזיר בקריאה ?what(a)  
 התשובה היא:

\_\_\_\_\_

(2 נק') (ii) אם נתון המערך a הבא:

|   |   |   |   |
|---|---|---|---|
| 0 | 1 | 2 | 3 |
| 1 | 2 | 1 | 4 |

איזה ערך השיטה תחזיר בקריאה ?what(a)  
 התשובה היא:

\_\_\_\_\_

(2 נק') (iii) אם נתון המערך a הבא:

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

איזה ערך השיטה תחזיר בקריאה ?what(a)  
 התשובה היא:

\_\_\_\_\_

(6 נק') (iv)

מה משמעותו של הערך המוחזר מהשיטה what בהינתן לה מערך מלא במספרים שלמים?  
 הסבר בקצרה.

\_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_

בהצלחה