האוניברסיטה הפתוחה

ט' באדר תשפ"ג

מס' שאלון - 468

2

במרץ 2023

סמסטר 2023א

מס' מועד 72

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 ארבע 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!
* אפשר לתעד בעברית. אין צורך בתיעוד API .

חומר עזר:
חוברות השקפים 1-6, 7-12.
אין להכניס חומר מודפס נוסף או חומר אחר מכל סוג.
אין להכניס מחשב או מחשבון או מכשיר אלקטרוני מכל סוג שהוא.
חומר העזר מודפס בלבד.

בהצלחה !!!

חלק א – כל התשובות צריכות להיכתב בתוך קובץ המבחן במקומות המתאימים בלבד. תשובה שתיכתב לא במקומה לא תיבדק.

שאלה 1 (25 נקודות)

נתונה מטריצה (מערך דו-ממדי matrix) ריבועית (מספר השורות שווה למספר העמודות והוא n) מלאה במספרים 0 ו-1 (בלבד).

המטריצה מתארת את ההיכרויות בקבוצה של n אנשים בדרך הבאה:

- לכל אדם בקבוצה יש מספר.
- אם אדם שמספרו i והאדם שמספרו j מכירים זה את זה, אזי במטריצה יש 1 במקומות [j][i] ו-[i][j]. אם הם לא מכירים זה את זה, אזי במטריצה יש 0 במקומות [j][i] ו-[i][j].
- המטריצה היא סימטרית ביחס לאלכסון הראשי. כלומר לא ייתכן שאדם אחד מכיר אדם שני, אך האדם השני לא מכיר את האדם הראשון.
- במקומות [i][i] במטריצה נמצא הערך 0.

כתבו שיטה סטטית **רקורסיבית** friends3 המקבלת מטריצת היכרויות שכזאת, והיא צריכה למצוא בתוכה מעגלי היכרות של שלושה אנשים: אדם שמספרו i שמכיר אדם שמספרו j ואדם שמספרו k, וגם האנשים j ו-k מכירים זה את זה.

דוגמאות:

- במטריצה הזאת:

	0	1	2	3	4	5	6	7	8	9
0	0	1	0	0	0	1	1	0	0	0
1	1	0	1	1	0	0	0	0	0	1
2	0	1	0	0	0	1	1	0	0	1
3	0	1	0	0	0	0	1	1	0	0
4	0	0	0	0	0	0	0	0	0	1
5	1	0	1	0	0	0	0	1	1	0
6	1	0	1	1	0	0	0	1	1	0
7	0	0	0	1	0	1	1	0	0	1
8	0	0	0	0	0	1	1	0	0	0
9	0	1	1	0	1	0	0	1	0	0

יש בדיוק שני מעגלי הכרות באורך 3 :

הראשון בין האנשים שמספריהם 1 2 ו-9 והשני בין האנשים שמספריהם 3 6 ו-7.

בדקו זאת על המטריצה כדי לוודא שהבנתם את השאלה.

- במטריצה הזאת (כולה מלאה ב- 1, כלומר כולם חברים של כולם):

	0	1	2	3
0	0	1	1	1
1	1	0	1	1
2	1	1	0	1
3	1	1	1	0

יש **ארבעה** מעגלי היכרות באורך 3:

1. 0 1 2
2. 0 1 3
3. 0 2 3
4. 1 2 3

עליכם לכתוב שיטה סטטית רקורסיבית שחתימתה

```
public static int friend3(int[][] mat)
```

המקבלת כפרמטר את מטריצת ההיכרויות, מדפיסה למסך את מספריהם של שלושת האנשים בכל מעגל היכרות, ומחזירה כמה מעגלי היכרות באורך 3 יש במטריצה.

בדוגמה הראשונה לעיל השיטה תדפיס על המסך

9 2 1

7 6 3

ותחזיר את הערך 2.

בדוגמה השניה לעיל השיטה תדפיס על המסך

0 1 2

0 1 3

0 2 3

1 2 3

ותחזיר את הערך 4.

שימו לב:

כל מעגל היכרות יש לספור ולהדפיס רק פעם אחת, ולא שלוש.

כל מעגל היכרות יש להדפיס בשורה נפרדת.

לא משנה מה הסדר הפנימי של מספרי האנשים בכל שורת הדפסה, ומה הסדר בין המעגלים (השורות).

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

אפשר להניח:

- **המערך מלא בנתונים חוקיים (כלומר רק 0 ו-1) והוא אינו null.**
- **המערך סימטרי סביב האלכסון הראשי, כלומר לכל i,j בגבולות המערך, מתקיים mat[i][j] == mat[j][i].**
- **באלכסון הראשי של המערך יש 0, כלומר לכל i בגבולות המערך מתקיים mat[i][i] = 0**

אין צורך לבדוק זאת!

אם אין מעגלי היכרות באורך 3 בכלל במערך, השיטה כמובן תחזיר 0 ולא תדפיס כלום.

- **אין צורך לדאוג ליעילות השיטה! אבל כמובן שצריך לשים לב לא לעשות קריאות רקורסיביות מיותרות!**
- **מי שיכתוב שיטה שרק תחזיר את מספר מעגלי ההיכרות ולא תדפיס את כל המעגלים יקבל 15 נקודות לכל היותר (אם הפתרון יהיה נכון).**
- **אל תשכחו לתעד את מה שכתבתם!**

שאלה 2 (25 נקודות)

נגדיר: בהינתן מערך חד-ממדי המלא במספרים שלמים, איבר שאינו קטן משני שכניו (מימין ומשמאל) נקרא **פסגה** (peak).

אם האיבר נמצא בקצה המערך, מתייחסים רק לשכן אחד.

דוגמאות:

- במערך {5, 10, 20, 15} יש פסגה אחת והיא המספר 20, שכן לאיבר 20 יש שני שכנים, 10 ו- 15, והוא גדול משניהם.
- במערך {10, 20, 15, 2, 23, 90, 67 } יש שני איברים שכל אחד מהם הוא פסגה. המספרים הם 20 ו- 90. 20 גדול משני שכניו 10 ו- 15, 90 גדול משני שכניו 23 ו- 67.
- במערך {10, 20, 15, 2, 90, 90, 67 } יש שלושה איברים שכל אחד מהם הוא פסגה. המספרים הם 20, 90 ו- 90. 20 גדול משני שכניו 10 ו- 15, 90 (השמאלי) אינו קטן משני שכניו 2 ו- 90, 90 (הימני) אינו קטן משני שכניו 90 ו- 67.
- במערך {1, 2, 3, 4, 5} יש פסגה אחת והיא האיבר 5 שהוא גדול מהשכן היחיד שלו 4.

כתבו שיטה סטטית המקבלת כפרמטר מערך חד-ממדי המכיל מספרים שלמים, ומחזירה ערך של פסגה שקיימת במערך. שימו לב שצריך להחזיר פסגה אחת, גם אם יש יותר מפסגה אחת במערך. לא משנה איזו פסגה תוחזר.

אפשר להניח שהמערך הנתון אינו null וכולל לפחות 2 איברים.

חתימת השיטה היא:

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

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.

ציינו מהי סיבוכיות זמן הריצה ומהי סיבוכיות המקום של השיטה שכתבתם. הסבירו תשובתכם.

אל תשכחו לתעד את מה שכתבתם!

חלק ב - את התשובות לשאלות 3 ו- 4 יש לכתוב על גבי השאלון. לא נבדוק תשובות שייכתבו במקום אחר!

שאלה 3 (25 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; }

     public void setNumber(int num)     { _number = num;    }
     public void setLeftSon(Node node)  { _leftSon = node;  }
     public void setRightSon(Node node) { _rightSon = node; }
}
```

המחלקה BinaryTree מאגדת בתוכה שיטות סטטיות לטיפול בעץ בינרי.

בין השיטות נתונות השיטות print ו- what הבאות:

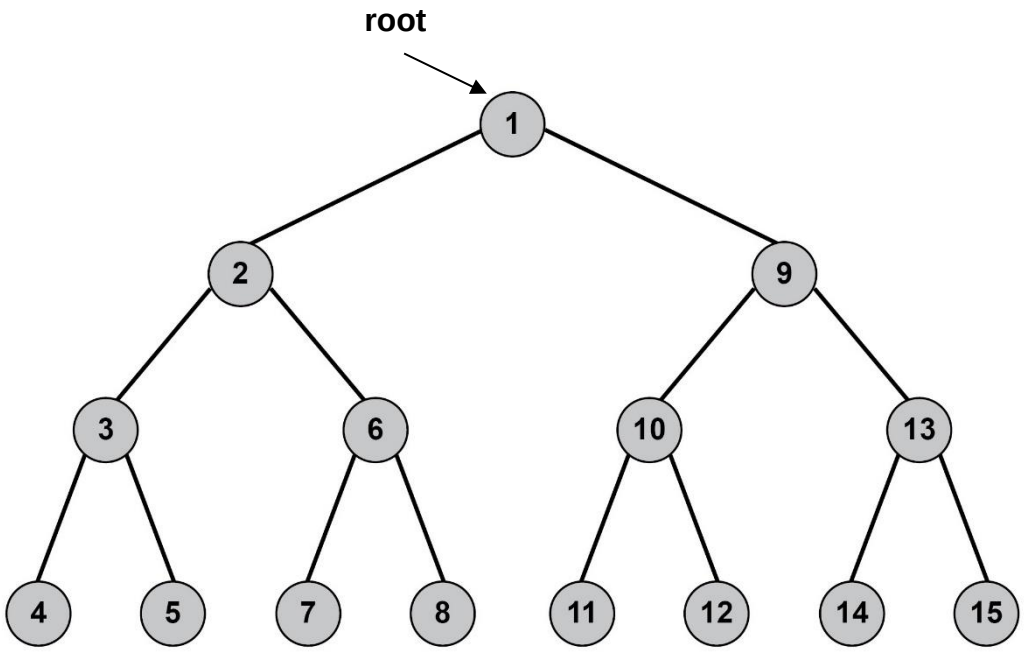
```
public static void print(Node root)
{
    if (root == null)
        return;
    System.out.print(root.getNumber());
    print(root.getLeftSon());
    print(root.getRightSon());
}
```

- המשך השאלה בעמוד הבא -

```
public static void what(Node root)
{
   if (root == null)
      return;
   what(root.getLeftSon(), root.getRightSon(), 1);
}

private static void what(Node root1, Node root2, int level)
{
   if (root1 == null || root2 == null)
      return;
   what(root1.getLeftSon(), root2.getRightSon(), level + 1);
   if (level %  2 == 1)
   {
       int temp = root1.getNumber();
       root1.setNumber(root2.getNumber());
       root2.setNumber(temp);
   }
   what(root1.getRightSon(), root2.getLeftSon(), level + 1);
}
```

נתון **העץ הבינרי** הבא, ששורשו הוא root



ענו על הסעיפים הבאים:

(א) מה תדפיס השיטה **print** בעקבות הקריאה BinaryTree.print(root) על העץ ששורשו root המצויר לעיל? (5 נק')

התשובה היא:

_____

_____

(ב) נקרא לשיטה BinaryTree.what(root) על העץ ששורשו root המצויר לעיל. איך ייראה העץ לאחר הפעלת השיטה? ציירו את העץ להלן בתוך המסגרת. (7 נק')

התשובה היא:

(ג) מה תדפיס השיטה **print** בעקבות הקריאה BinaryTree.print(root)? **(לאחר ביצוע סעיף ב)** (5 נק')

התשובה היא:

_____

_____

המשך הבחינה בעמוד הבא

(ד) אם נשנה בתוך השיטה what את הפקודה

```
if (level % 2  == 1)
```

כך שהיא תהיה

```
if (level % 2  == 0)
```

ונקרא לשיטה what המעודכנת BinaryTree.what(root) על העץ ששורשו root המצויר בראש השאלה. איך ייראה העץ לאחר הפעלת השיטה? ציירו את העץ להלן בתוך המסגרת. שימו לב, הפעלת השיטה נעשית על העץ המקורי. (8 נק')

התשובה היא:

המשך הבחינה בעמוד הבא

שאלה 4 (25 נקודות)

נתון פרויקט שהוגדרו בו המחלקות A ו- B שלהלן. כל אחת בקובץ נפרד, כמובן.

```
public class A
{
    private int _n;
    private char _ch;

    public A()              { _n = 2;        _ch = 'X'; }
    public A(int n)         { _n = n;        _ch = 'Y'; }
    public A(int n, char ch) { _n = n;        _ch = ch; }
    public A(A other)       { _n = other._n; _ch = other._ch;}

    public int getN()       { return _n;                }
    public char getCh()     { return _ch;               }

    public void inc()       { _n++;          _ch++;     }
    public String toString()
    {
        String s = "";
        for (int i=0; i<_n; i++)
            s+= _ch;
        return s;
    }

} //end of class A


//-------------------------------------------------------------

public class B extends A
{
    private A _a;

    public B()              { super();    _a = new A();  }
    public B(int n)         { super(n);   _a = new A();  }
    public B(A other)       { super();    _a = new A(other); }
    public B(A other,int n) { super(other); _a = new A(n);   }

    public void inc()        { _a.inc();  }
    public String toString() { return _a.toString();    }
```

- המשך המחלקה בעמוד הבא -

```
    private int comp (int n, int m)
    {
        if (n > m)
        return n;
        return m;
    }

    private char comp (char ch1, char ch2)
    {
        if (ch1 < ch2)
        return ch1;
        return ch2;
    }

    public A makeA()
    {
        return new A(comp(_a.getN(), getN()),
                     comp(_a.getCh(), getCh()));
    }

} //end of class B
```

כמו כן נתונה המחלקה Driver הבאה באותו פרויקט:

```
public class Driver
{
    public static void main (String[] args)
    {
            // כאן יופיעו קטעי הקוד שיש בסעיפים להלן
    }
}
```

עליכם לענות על הסעיפים הבאים. הניחו שהם מופעלים אחד אחרי השני.

בכל סעיף עליכם לכתוב מה יקרה כתוצאה מהרצת קטע הקוד. אם תהיה שגיאת קומפילציה או הרצה, כתבו איזו שגיאה ונמקו. אם הקוד תקין, כתבו מה יודפס על הפלט.

סעיף א (4 נקודות) –

נניח שנתון קטע הקוד הבא בתוך השיטה main במחלקה Driver:

```
A a1 = new A(3, 'C');
System.out.println(a1);
```

התשובה היא:

_____

_____

סעיף ב (4 נקודות) –

נניח שנתון קטע הקוד הבא בתוך השיטה main במחלקה Driver:

```
A a2 = new A(5);
System.out.println(a2);
```

התשובה היא:

_____

_____

סעיף ג (4 נקודות) –

נניח שנתון קטע הקוד הבא בתוך השיטה main במחלקה Driver:

```
B b1 = new B(a1);
a1.inc();
System.out.println(b1);
System.out.println(a1);
```

התשובה היא:

_____

_____

סעיף ד (4 נקודות) –

נניח שנתון קטע הקוד הבא בתוך השיטה main במחלקה Driver:

```
B b2 = new B(b1, 5);
System.out.println(b2);
```

התשובה היא:

_____

_____

סעיף ה (4 נקודות) –

נניח שנתון קטע הקוד הבא בתוך השיטה main במחלקה Driver:

```
A a3 = new B(4);
System.out.println(a3);
```

התשובה היא:

_____

_____

סעיף ו (5 נקודות) –

נניח שנתון קטע הקוד הבא בתוך השיטה main במחלקה Driver:

```
A a4 = b2.makeA();
System.out.println(a4);
```

התשובה היא:

_____

_____

ב ה צ ל ח ה!