

## מבני נתונים - סמסטר קיץ תש"פ

### מטלה 2

#### הנחיות:

- מטלה זו הינה להגשה ביחידים. העתקה תגרור **לפסילה מלאה** של המטלה למעתיק והמועתק.
- בכל השאלות שאינן רק תיאורטיות במטלה זו ניתן לכתוב תשובות בפסאודו-קוד או ב-JAVA, במקרה שכתבתם ב-JAVA העתיקו את הקוד לקובץ הפתרון, כך שכל פתרון המטלה יהיה מוכל בקובץ יחיד, PDF או תמונה, ללא הגבלה על מספר העמודים. יש להגיש את המטלה בקובץ ZIP (ולא כל דחיסה אחרת) המכיל את הקובץ הנ"ל בלבד. שם קובץ ה-ZIP יהיה מספר ת.ז. של התלמיד. אין להגיש קבצים או תיקיות מיותרים. סטייה מהנחיות אלו תגרור הורדה בציון.
- בשאלה 2 יש סעיף בונוס של 5 נקודות, יחד עם זאת ציון המטלה כולו אינו יכול לעלות על 100.

#### שאלה 1:

בשאלה זו יש למצוא את מספר איים במטריצה שמורכבת מאפסים ואחדות. קבוצה של אחדות מהווה אי, כאשר יש להם צלע משותף או קדקוד משותף (חיבור באלכסון). יש לעבור על כל אלמנט בכל מערך פעם אחת בלבד.

דוגמה. קלט:

```
{ { 1, 1, 0, 0, 0 },
  { 0, 1, 0, 0, 1 },
  { 1, 0, 0, 1, 1 },
  { 0, 0, 0, 0, 0 },
  { 1, 0, 1, 1, 0 } }
```

פלט: 4.

#### שאלה 2:

יש להוסיף למחלקה המצורפת פונקציה שמקבלת עץ בינארי ומחזירה true אם הוא עץ חיפוש בינארי, אחרת היא מחזירה false:

```
public static boolean isBST(BinaryTree bt){ . . . }
```

**שאלה 3:**

נגדיר עץ BVL כעץ בינארי שמקיים את התכונה שלכל צומת, ההפרש בין הגבהים של הילדים של צומת הוא לכל היותר 2. הוכיחו שעץ BVL הוא עץ מאוזן.

**שאלה 4**

(א) מדוע לעץ אדום-שחור לא יכול להיות צומת שחור עם ילד שחור אחד בדיוק;

(ב) האם כל תת-עץ של עץ אדום-שחור הוא בעצמו עץ אדום שחור?  
אם כן – יש להוכיח את הטענה, אם לא – יש לתת דוגמה נגדית.

(ג) נניח שהשורש של עץ אדום-שחור הוא אדום, אבל שאר התכונות של עץ אדום-שחור נשמרות. אם נעשה אותו שחור, האם העץ נשאר עץ אדום-שחור?  
אם כן – יש להוכיח את הטענה, אם לא – יש לתת דוגמה נגדית.

**שאלה 5**

בשאלה זו יש לכתוב פונקציה

```
public static void report(String[] sArr){...}
```

**קלט:** מערך של מילים (של Strings).

**פלט:** בסוף הריצה על התוכנית יש להדפיס דוח שמכיל את הפרטים הבאים:

- מספר המילים הכללי.
- מספר המילים השונות.
- מספר המילים שמופיעות במערך יותר מפעם אחת.
- המילה שחוזרת הכי הרבה פעמים ואת מספר הפעמים שהיא חוזרת על עצמה.
- את המילה הארוכה ביותר.

**יש לפתור את הבעיה במעבר אחד בלבד על מערך המילים.**

**בהצלחה רבה!**

```

class Node{
    String data;
    Node left, right;

    Node(String data){
        this.data = data;
        this.left = null;
        this.right = null;
    }
    public String toString(){return ""+data;}
    public String getData() {return data;}
    public Node getLeft() {return left;}
    public Node getRight() {return right;}
} // Node

public class BinaryTree {
    private Node root;

    public BinaryTree() {this.root = null;}

    public static boolean isBST(BinaryTree bt) {. . . }
}

```