

## מבוא לחישוב 2-7015710 סמסטר א'

קבוצות: 1,4,5,7,8,9

### מבחן מועד א' - 3.2.20

ליעד גוטליב, רויטל מרבל, סעיד עסלי.

- משך המבחן: 2.5 שעות.
- מחברת שורות. אין שימוש בחומר עזר.
- יש להחזיר את דף המבחן בסוף המבחן.
- במבחן חמש שאלות, כולם חובה.
- בכל שאלה ניתן לכתוב פונקציות עזר כרצונכם.

#### שאלה 1 (20 נקודות)

**למספר אדיר** (powerful number) יש את התכונה הבאה: אם הוא מתחלק במספר ראשוני  $p$ , אזי הוא גם מתחלק בריבוע  $p^2$ . לדוגמה, 72 הוא מספר אדיר, שכן הוא מתחלק במספרים הראשוניים 2,3 (בלבד) וגם בריבוע שלהם 4,9. כתבו פונקציה שבהנתן מספר שלם, מחזירה true אך ורק אם המספר הוא מספר אדיר:

**public static boolean powerful(int n)**

#### שאלה 2 (20 נקודות)

מחרוזת נקראת מאוזנת אם היא אינה ריקה, והאותיות R ו-S מופיעות בה אותו מספר פעמים.

מחרוזת t היא קידומת של s, אם t הוא תת-מחרוזת בתחילת s. לדוגמה, הקידומות של "java" הם: "j", "ja", "jav", "java".

כתבו פונקציה שמקבלת מחרוזת, ומחזירה את כמות הקידומות המאוזנות שאפשר לגזור מהמחרוזת הנתונה. ניתן להניח שבמחרוזת ישנם רק את האותיות R, S.

**public static int balanced(String s)**

#### דוגמאות:

"SRSR" → 3 ("SR", "SRSR")

"RSSRS" → 2 ("RS", "RSSR")

### שאלה 3 (20 נקודות)

כתבו פונקציה המקבלת מערך דו-ממדי ומדפיסה אותו בצורת ספירלה. לדוגמא עבור המטריצה  $\{\{1,2,3,4\}, \{12,13,14,5\}, \{11,16,15,6\}, \{10,9,8,7\}\}$ , הפונקציה תדפיס:

1	2	3	4
12	13	14	5
11	16	15	6
10	9	8	7

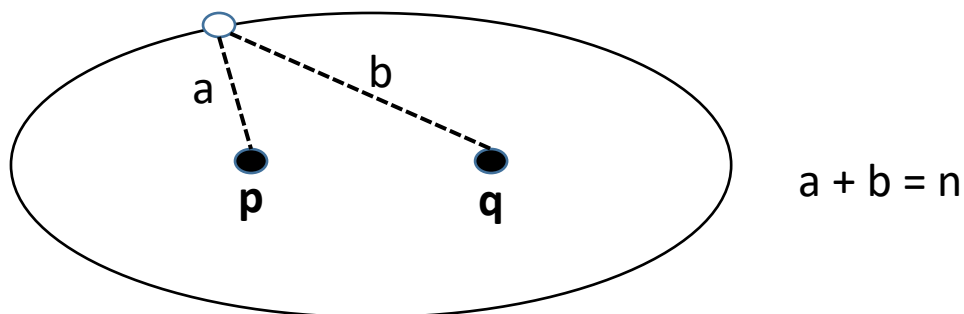
`public static void spiral(int[][] a)`

ניתן להניח שהמערך הוא ריבועי.

**הכוונה:** חישבו קודם איך מדפיסים את המסגרת של המערך.

### שאלה 4 (20 נקודות)

יש לכתוב מחלקה בשם **Ellipse** (אליפסה). האליפסה מוגדרת על ידי שתי נקודות מרכז  $p, q$  ומספר ממשי  $n$ . הקוד לנקודה נמצא בנספח, וחובה להתייחס רק אליו ללא שינויים.



1. כתבו את כותרת המחלקה ואת השדות (המשתנים) שלה.

2. כתבו שני בנאים: בנאי שמקבל חמישה נתונים **בסדר** (משמאל לימין)  $x_p, y_p, x_q, y_q, n$ , ובנאי מעתיק שמקבל אליפסה חדשה. בנאי המעתיק ישתמש בהעתקה עמוקה.

3. הוסיפו למחלקה שיטה שמקבלת נקודה ומחזירה 0 אם הנקודה נמצאת על שפת האליפסה, 1- אם היא נמצאת בתוך האליפסה, או 1 אם היא נמצאת מחוץ לאליפסה. נקודה נמצאת על שפת האליפסה אם סכום של המרחק שלה משתי נקודות המרכז שווה ל- $n$ . אם הסכום יותר מ- $n$ , אזי היא מחוץ לאליפסה, ואם הסכום פחות מ- $n$  אזי היא בפנים (כמו באיור למעלה).

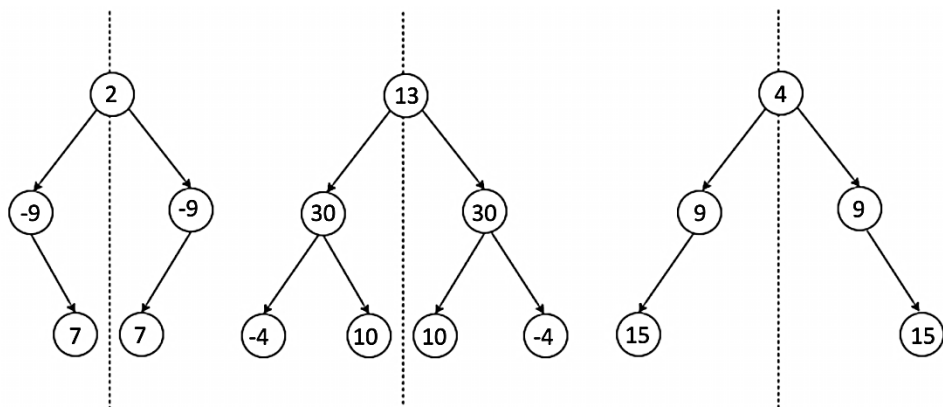
`public int where(Point p){...}`

4. הוסיפו למחלקה שיטה להשוואת שתי אליפסות. (שני אליפסות שווים אם הנקודות שלהם זהות, וגם הערכים ח זהים.) השיטה מחזירה true אם האליפסות שווים, אחרת היא מחזירה false.

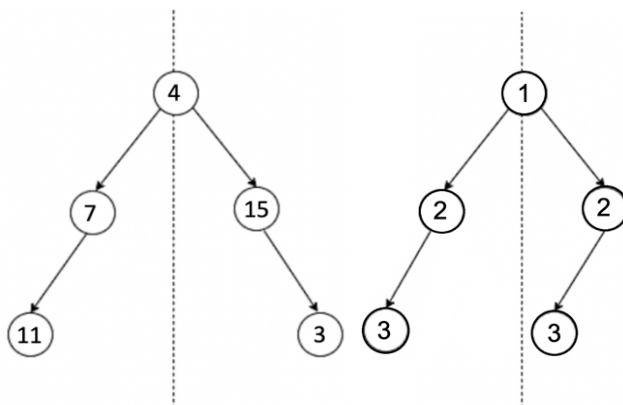
```
public boolean equals(Ellipse e){...}
```

### שאלה 5 (20 נקודות)

עץ בינארי נקרא סימטרי אם הוא סימטרי ביחס לשורש שלו, ז"א הצד הימני של השורש הינו תמונת מראה של הצד השמאלי. דוגמאות לעצים סימטריים:



דוגמאות לעצים לא סימטריים:



כתבו פונקציה שמקבלת שורש של עץ בינארי, ומחזירה true אם העץ הוא סימטרי, ואחרת false. הקוד לצומת מופיע בנספח, וחובה להתייחס רק אליו ללא שינויים.

```
public static boolean isSymmetric(BTNode root)
```

```

public class Point {
    private double xVal;
    private double yVal;

    public Point(double x, double y) {
        xVal = x;
        yVal = y;
    }

    public double distance(Point p) {
        double xDis = xVal - p.getX();
        double yDis = yVal - p.getY();
        return Math.sqrt(xDis*xDis + yDis*yDis);
    }

    public boolean equals(Point p) {
        Return xVal==p.xVal && yVal==p.yVal;
    }
}

```

```

public class BTNode {
    private int data;
    private BTNode left, right;

    public BTNode(int data) {
        this.data = data;
    }

    public int getData() {
        return data;
    }

    public BTNode getLeft() {
        return left;
    }

    public void setLeft(BTNode left) {
        this.left = left;
    }

    public BTNode getRight() {
        return left;
    }

    public void setRight(BTNode right) {
        this.right = right;
    }
}

```

