

מבוא לחישוב 2-7015710 סמסטר א'

קבוצות: 1,4,5,7,8

מבחן מועד א' - 3.2.19

ליעד גוטליב, אודי לביא, גיל לוי, 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)

מספר מרסן (Marsenne number) הוא מספר שהוא חזקה של 2 פחות 1, כלומר בתבנית $2^i - 1$ עבור כל מספר שלם חיובי i . לדוגמה 1,3,7,15,31,63,127,255 הם כולם מספרי מרסן.

מספר מרסן ראשוני (Marsenne prime) הוא גם ראשוני וגם מספר מרסן. לדוגמה, המספרים 3,7,127 הם מספרי מרסן ראשוניים.

כתבו פונקציה שבהנתן מספר שלם, מחזירה true אם המספר הוא מספר מרסן ראשוני:

```
public static boolean MarsennePrime(int n)
```

שאלה 2 (20 נקודות)

כתבו פונקציה בשם single שמקבלת מחרוזת s ומחזירה מחרוזת אשר מופיעים בה רק תווים המופיעים פעם אחת ב-s. סדר האותיות במחרוזת החדשה יהיה זהה לסדרם ב-s. אם אין תו המופיע פעם אחת ב-s, יש להחזיר מחרוזת ריקה.

```
public static String single(String s)
```

דוגמה. קלט: "abcd9cb"

פלט: "ad9"

דוגמה. קלט: "abracadabra"

פלט: "cd"

שאלה 3 (20 נקודות)

מטריצה תקרא "מטריצה אופקית" אם היא בעלת מספר שורות זוגי, ומתקיים בה התנאי שבכל עמודה סכום האברים שבמחצית השורות העליונות שווה לסכום האברים שבמחצית השורות התחתונות.

לדוגמא: $\text{int}[][] a = \{\{7,10,8\},\{2,-3,0\},\{4,1,2\},\{5,6,6\}\}$ היא מטריצה אופקית.

7	10	8
2	-3	0
4	1	2
5	6	6

כתבו פונקציה המקבלת מטריצה ומחזירה אמת אך ורק אם היא אופקית.

public static boolean horizontal(int[][] a)

שאלה 4 (20 נקודות)

כתבו מחלקה בשם **Student** המייצגת סטודנט באוניברסיטה.

1. הוסף למחלקה שלוש שדות (משתנים) פרטיים, המיצגים את שם התלמיד, גילו, וממוצע ציונים של התלמיד. הממוצע יהיה מספר ממשי.
2. כתבו למחלקה שלשה בנאים: בנאי המקבל שלשה נתונים, בנאי המקבל שני נתונים ומאתחל את הממוצע להיות 0, ובנאי מעתיק.
3. כתבו למחלקה שיטות set ו-get לכל שדה.
4. כתבו שיטת toString() המחזירה תיאור נתוני האובייקט.
5. כתבו את השיטה equals אשר מקבלת כפרמטר סטודנט אחר ומחזירה true או false לאחר השוואה של גיל וממוצע.

שאלה 5 (20 נקודות)

מצורפת למבחן מחלקת **MyLinkedList** המייצגת רשימה מקושרת חד-כיוונית. יש להוסיף למחלקה פונקציה

public boolean invert()

ההופכת את סדר הצמתים ברשימה. לדוגמה, אם סדר הצמתים היה (2,5,1), הסדר החדש יהיה (1,5,2).

שימו לב: השינוי היחיד שניתן לעשות במחלקה LinkedList הוא הוספת הפונקציה המבוקשת (וגם הוספת פונקציות עזר). לא ניתן לשנות שדה או פונקציה הנמצאים במחלקה, או להוסיף שדות למחלקה. לא ניתן לעשות כל שינוי במחלקה Node.

נספח: רשימה מקושרת:

```
public class Node {
    private String data;
    Node next;
    public Node(String data){
        this.data = data;
        this.next = null;
    }
    public String toString(){
        return data;
    }
}

public class MyLinkedList {
    private Node head;
    private int size;

    public MyLinkedList(){
        head = null;
        size = 0;
    }

    public void add(String data){
        if (head == null){
            head = new Node(data);
        }
        else{
            Node n = head;
            while(n.next != null){
                n = n.next;
            }
            n.next = new Node(data);
        }
        size++;
    }

    public String toString(){
        String ans = "[";
        if (head == null) ans = "[]";
        else {
            Node n = head;
            while(n.next != null){
                ans = ans + n + ", ";
                n = n.next;
            }
            ans = ans + n + "];"
        }
        return ans;
    }
}
```