

הפקולטה למדעי הטבע - המחלקה למדעי המחשב

מסטר א', מועד ב', יום שני ו' אדר תשע"ו, Monday, February 15, 2016

מחלקה: מדעי המחשב

מרצה: פרופ' ואדים (דוד) לויט, גב' אליזבת איצקוביץ

שם הקורס: אלגוריתמים 1

מס' הקורס: 7027010

משך הבחינה: 3 שעות

חל איסור על שימוש בכל חומר עזר.

הנחיות כלליות:

- המבחן ייבדק בצורה אוטומטית ע"י תוכנת מחשב שתשתמש בשמות המוזכרים להלן.
- יש לרשום אלגוריתמים, סיבוכיות והוכחות בקובץ **word** בשם **algorithms**.
- יש לדחוס את קובץ **word** וכל קבצי **java** בקובץ אחד. שם הקובץ – מספר תעודת זהות, סוג הקובץ - **rar** או **zip**.
- שפת תכנות – **java**.

הנחיות לתכנות:

- יש לבנות **java project** בשם **Exam**
- לפתרון של כל שאלה צריך לבנות מחלקה נפרדת.
- את קובץ הפרויקט ואת השאלון ניתן להוריד מ"מתזמן מבחנים".

**המבחן שלא יעמוד בדרישות אלו לא ייבדק!**

הפקולטה למדעי הטבע - המחלקה למדעי המחשב

$$20 + 20 = 40$$

בעיה מס' 1

20

(א) נתונה מטריצה המורכבת מאפסים ואחדות בלבד. יישמו את אלגוריתם המחשב את גודל של התת-מטריצה הריבועית המקסימאלית המורכבת מאחדות בלבד או מאפסים בלבד.

**דוגמה 1 :** גודל של התת-מטריצה הריבועית המקסימאלית המורכבת מאחדות בלבד או מאפסים בלבד שווה  $3 \times 3 = 9$  והיא מורכבת מאחדות בלבד

1	0	1	0	0
1	1	1	1	0
0	1	1	1	1
1	1	1	1	0
1	1	1	0	0

**דוגמה 2 :** גודל של התת-מטריצה הריבועית המקסימאלית המורכבת מאחדות בלבד או מאפסים בלבד שווה  $2 \times 2 = 4$  והיא מורכבת מאפסים בלבד:

1	0	1	1
1	1	0	0
1	0	0	1
1	0	0	0

(ב) נתונה מטריצה המורכבת מאפסים ואחדות בלבד. יישמו את אלגוריתם המחשב את גודל של התת-מטריצה המלבנית המקסימאלית (בעלת שטח גדול ביותר) המורכבת מאחדות בלבד. שטח של תת-מטריצה מלבנית מוגדר ככפל של מספר שורות שלה במספר עמודות שלה.

**דוגמה: קלט:**

1	1	1	0	1
1	1	1	1	0
0	1	1	1	0
0	1	1	0	0

**פלט :** גודל התת-מטריצה המלבנית בעלת שטח גדול ביותר שווה  $2 \times 4 = 8$

**בעיה 1 - המשך**

יש להכין מחלקה בשם Q1.

בתוך המחלקה יש לכתוב שתי פונקציות סטטיות:

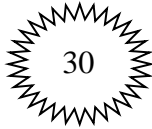
**א)** פונקציה שמקבלת מטריצה המורכבת מאפסים ואחדות בלבד ומחזירה את גודל של התת-מטריצה הריבועית המקסימאלית המורכבת מאחדות בלבד או מאפסים בלבד:

```
public static int maxSquareSubMatrix(int[][] mat){...}
```

**ב)** פונקציה שמקבלת מטריצה המורכבת מאפסים ואחדות בלבד ומחזירה את גודל של התת-מטריצה המלבנית המקסימאלית (בעלת שטח מקסימאלי) המורכבת מאחדות בלבד:

```
public static int maxRectangularSubMatrix(int[][] mat){...}
```

אלגוריתם, סיבוכיות, דוגמא והוכחות.



## בעיה מס' 2

**פלינדרום** הוא רצף סמלים, שאם נתעלם מרווחים בין המילים יהיה ניתן לקרוא אותו משני הכיוונים, משמאל לימין ומימין לשמאל, ללא שינוי בתוצאה. למשל, משפט "WAS IT A CAR OR A CAT I SAW" הוא משפט פלינדרומי.

יש ליישם את האלגוריתם המחשב אחת מתתי-מחרוזות **הפלינדרומיות הארוכות ביותר** של המחרוזת הנתונה.

### דוגמה 1:

קלט: "alfalfa"

פלט: אחת מתתי מחרוזות הפלינדרומיות הארוכות ביותר היא "alala"  
(יש עוד מס' תתי-מחרוזות פלינדרומיות ארוכות ביותר, למשל, " alfla " ועוד...)

**דוגמה 2:** מחרוזות שמורכבת מאות אחת גם מהווה פלינדרום.

קלט: "a"

פלט: "a"

יש להכין מחלקה בשם **Q2**.

בתוך המחלקה יש לכתוב פונקציה סטטית שמקבלת מחרוזת תווים ומחזירה אחת מתתי-מחרוזות הפלינדרומיות הארוכות ביותר

```
public static String lps(String s){...}
```

אלגוריתם, סיבוכיות, דוגמא והוכחות.

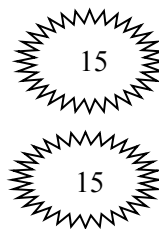
הפקולטה למדעי הטבע - המחלקה למדעי המחשב

$$15 + 15 = 30$$

**בעיה מס' 3:** יישמו את האלגוריתם המחשב את עלות המסלול הזול ביותר, ואת עלות המסלול בעל עלות הבאה בגודלה (הנקראת עלות משנית) בין שני קודקודים מסוימים בדף חשבוני בגודל  $M$  על  $N$  (עם משקלים מונחים על הצלעות). בנוסף, בדף זה ישנו אזור הנקרא "שטח מת", שאף מסלול לא יכול לעבור בו. צורתו של השטח הוא אוסף (מערך) של נקודות.

**קלט:** מערך דו-ממדי של קודקודים. כל קודקוד (Node) מכיל את המשקלים של שתי הצלעות היוצאות ממנו לכיוונים ימינה ומעלה:

x - משקל המונח על הצלע האופקית, (הכיוון, כמו בציר ה-X, ימינה).  
y - משקל המונח על הצלע האנכית, (הכיוון כלפי מעלה, כמו בציר ה-Y).  
שתי הנקודות שביניהן מחשבים את המסלולים הזולים ביותר הן  $(0,0)$  ו- $(M,N)$ .  
"שטח מת" - מערך של נקודות שאף מסלול לא יכול לעבור בהן.



**פלט:**

(א) עלות המסלול הזול ביותר.  
(ב) עלות המסלול בעל עלות משנית.

**דוגמה:**

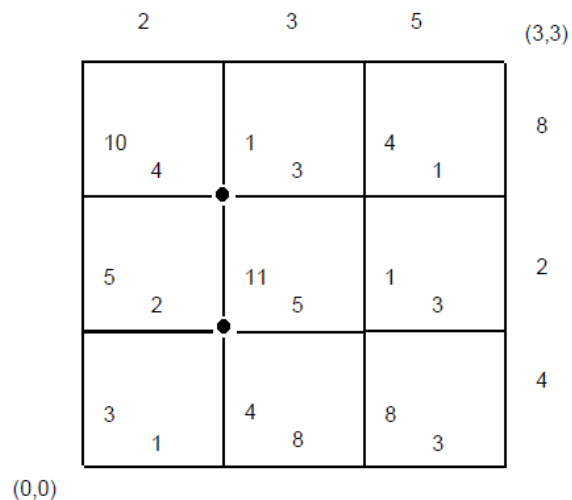
**קלט:** במערך דו-ממדי שלהלן:

קודקוד  $(0,0)$  נראה כך:  $x = 1; y = 3;$

קודקוד  $(0,3)$  נראה כך:  $x = 0; y = 4;$

קודקוד  $(3,3)$  נראה כך:  $x = 0; y = 0;$

Point forbidden [ ] = {new Point(1,1), new Point(2,1)}



**פלט:** עלות המסלול הזול ביותר 26, עלות המסלול בעל עלות משנית 27.

הפקולטה למדעי הטבע - המחלקה למדעי המחשב

יש להכין מחלקה בשם Q3.  
יש לכתוב בנאי המחלקה ושתי שיטות.

כמובן, ניתן לכתוב שיטות עזר לפי הצורך.

1. בנאי המחלקה מקבל משקלים כמערך דו-ממדי של קדקודים ומערך של נקודות:

```
public Q3(Node[][] nodes, Point forbidden[]){...}
```

מבנה של קדקוד אחד מיוצג ע"י מחלקת Node:

```
class Node{
    int x, y
    public Node(int x, int y){
        this.x = x;
        this.y = y;
    }
}
```

2. שיטה שמחזירה את עלות המסלול הזול ביותר:

```
public int cheapestPathPrice(){...}
```

3. שיטה שמחזירה את העלות המשנית:

```
public int secondCheapestPathPrice () {...}
```

הקוד שמבצע מילוי המערך דו-ממדי שבדוגמה ומחלקת Point נמצאים בעמודים הבאים.

אלגוריתם, סיבוכיות, דוגמא והוכחות.

```
public static Node[][] initMatOfNodes(){ // n = 4
    int n=4;
    Node mat[][] = new Node[n][n];
    //the 1-st row
    mat[0][0] = new Node(1,3);
    mat[0][1] = new Node(8,4);
    mat[0][2] = new Node(3,8);
    mat[0][3] = new Node(0,4);
    //the 2-nd row
    mat[1][0] = new Node(2,5);
    mat[1][1] = new Node(5,11);
    mat[1][2] = new Node(3,1);
    mat[1][3] = new Node(0,2);
    //the 3-d row
    mat[2][0] = new Node(4,10);
    mat[2][1] = new Node(3,1);
    mat[2][2] = new Node(1,4);
    mat[2][3] = new Node(0,8);
    //the 4-th row
    mat[3][0] = new Node(2,0);
    mat[3][1] = new Node(3,0);
    mat[3][2] = new Node(5,0);
    mat[3][3] = new Node(0,0);
    return mat;
}
```

```
public class Point{
    private int x, y;
    public Point(int x, int y){
        this.x = x;
        this.y = y;
    }
    public Point(){
        x = 0;
        y = 0;
    }
    public int x(){
        return x;
    }
    public int y(){
        return y;
    }
    public String toString(){
        return "("+x+", "+y+")";
    }
}
```

**בהצלחה!**