

## מבוא למדעי המחשב 202-1-101-1

סמסטר א' תשס"ד

מבחן מועד א'

מר בועז בן-משה  
פרופ' דניאל ברנד  
פרופ' מיכאל קודיש  
דר' יצחק רוזן

משך המבחן שעתיים וחצי – לא תינתן הארכה.

חומר עזר - אסור  
אין להשתמש במחשבון.

במבחן זה 4 שאלות, בניקוד המסתכם ב- 100 נקודות. ענו על כל השאלות.

אנא רשמו את תשובותיכם בדף התשובות בלבד. המחברת שקיבלתם היא **מחברת טיוטה ולא תימסר כלל לבדיקה**. בסיום הבחינה נשמור אך ורק את דף התשובות. כל שאר החומר יועבר לגריסה. הקפידו לרשום בדף התשובות גם את **מספר הנבחן ומספר החדר בו אתם נבחנים**.

בשאלות התכנות, מספר השורות העומדות לרשותכם בדף התשובות רומז על אורך הקוד הנדרש. הקפידו על כתב יד ברור. תשובות מסורבלות או ארוכות מדי לא תזכינה בניקוד מלא. אין צורך להעתיק את שורות הקוד הנתונות בשאלון לדף התשובות.

**בהצלחה!**



### שאלה 1 ( 25 נקודות)

השיטה הבאה מקבלת מערך של מספרים שלמים. על השיטה להחזיר ערך true אם ורק אם המערך מכיל איבר המופיע פעם אחת בלבד.  
לדוגמא:

מערך המספרים	ערך מוחזר
{4,3,2,4,5,2}	true
{4,2,4,2}	false
{4}	true
{}	false

ניתן להניח כי הפרמטר המתקבל הוא מערך שאינו null של מספרים שלמים.  
השלימו את קטע הקוד הבא כך שיבצע את הנדרש:

```
static boolean single(int[] arr) {  
    boolean ans = false;  
    // fill in  
    return ans;  
}
```

## שאלה 2 (24 נקודות)

נתונות שתי המחלקות הבאות המבצעות, בין השאר, 6 הדפסות למסך.  
ציין בדף התשובות מה תדפיס הקריאה לשיטה main.  
כל שורת הדפסה נכונה תזכה ב- 4 נקודות. כל שורת הדפסה לא נכונה תוריד 4 נקודות.  
שורה ריקה לא תוריד ולא תוסיף ניקוד. (הקפד להשאיר שורה ריקה בדף התשובות אם בחרת לא לכתוב שורת הדפסה).

```
1 public class A{
2     public void f(){System.out.println("A.f");}
3     public void g(){f();}
4 }
5
6 public class B extends A{
7     public void f(){System.out.println("B.f");}
8     public void g(){System.out.println("B.g");}
9     public void superG(){super.g();}
10
11     public static void main(String[] args ){
12         B b = new B();
13         A x = new B();
14
15         x.g();
16         b.g();
17
18         ((B)x).superG();
19         b.superG();
20
21         ((A)x).g();
22         ((A)b).g();
23     }
24 }
```

### שאלה 3 (20 נקודות)

הגדרה:

חלוקה אי-זוגית של מספר שלם חיובי  $n$  היא סדרה לא יורדת (סידרה שאבריה עולים או שווים) של מספרים אי-זוגיים שסכומם  $n$ .  
בשאלה זו נעסוק בשיטה רקורסיבית, המקבלת מספר שלם  $N$  ומדפיסה על המסך את כל החלוקות האי-זוגיות של  $n$ . להלן דוגמאות עבור הקלטים  $n=6$  ו- $n=7$ .

$n = 6$	$n = 7$
1 1 1 1 1 1	1 1 1 1 1 1 1
1 1 1 3	1 1 1 1 3
3 3	1 3 3
1 5	1 1 5
	7

נתונה השיטה:

```
static void oddPartitions(int n)
{oddPartitions(n, n, "");}
```

כמו כן נתונה השיטה הבאה. שיטה זו מקבלת שלושה משתנים ופרמטרים: מספר שלם חיובי  $n$ , מספר שלם חיובי  $max$  ומחרוזת  $suffix$ . השיטה מדפיסה את כל החלוקות האי-זוגיות של  $n$  שבהן כל המספרים המשתתפים בחלוקה אינם גדולים מ- $max$  ולכל חלוקה מצורפת בסופה המחרוזת  $suffix$ . לדוגמא, עבור הקריאה

```
oddPartitions(7, 3, "cat");
```

יתקבל הפלט הבא:

```
1 1 1 1 1 1 1 cat
1 1 1 1 3 cat
1 3 3 cat
```

השלימו את קטע הקוד הבא כך שיבצע את הנדרש:

```
static void oddPartitions(int n, int max, String suffix)
{
    if(n == 0)
        System.out.println(suffix);
    else
        // fill in
}
```

## שאלה 4 (31 נקודות)

**הגדרה:** עץ בינארי הינו עץ חיפוש בינארי אם הערך בכל צומת גדול מהערכים שבצמתים שבתת העץ השמאלי של הצומת וגם קטן מהערכים שבצמתים שתת העץ הימני של הצומת. שימו לב כי מהגדרה זו נובע כי הערכים בקודקודים של עץ חיפוש בינארי שונים זה מזה.  
בשאלה זו נתייחס לעץ בינארי שבקודקודיו יש ערכים שלמים (int).  
נתונות המחלקות עץ בינארי (BinaryTree) וצומת בעץ בינארי (BTN). (לכל צומת יש ערך שלם):  
למחלקה עץ בינארי השיטות הבאות:

```
public class BinaryTree {
    boolean isEmpty() {...}           // מחזירה אמת אם ורק אם העץ ריק
    BTN get_root(){...}              // מחזירה מצביע לצומת השורש
    BinaryTree left(){...}           // מחזיר את תת העץ השמאלי
    BinaryTree right(){...}          // מחזיר את תת העץ הימני
    int max();{...}                  // מחזירה את הערך הגדול ביותר בעץ
}

// למחלקה צומת של עץ בינארי השיטות הבאות:

public class BTN {
    BTN left_node(){...}             // מחזירה את צומת הבן השמאלי
    BTN right_node(){...}           // מחזירה את צומת הבן הימני
    int data(){...}                 // מחזירה את הערך של הצומת
}
```

## סעיף א' (10 נקודות)

במחלקה BinaryTree, כתבו את השיטה `int min()`, המחזירה את הערך המינימאלי בעץ שאיננו ריק.  
(ניתן להניח כי השיטה תיקרא רק עבור עץ שאינו ריק.)

## סעיף ב' (21 נקודות)

במחלקה BinaryTree, כתבו את השיטה `boolean isBST()`, הבודקת האם העץ הינו עץ חיפוש בינארי.  
(עץ ריק הוא עץ חיפוש בינארי.)

מבוא למדעי המחשב 202-1-101-1  
מבחן מועד א' - סמסטר א' תשס"ד - דף תשובות

תדר: \_\_\_\_\_ מספר נבחן: \_\_\_\_\_

זכרו: מספר השורות העומדות לרשותכם רומז על אורך הקוד הנדרש. אין צורך להעתיק את חלקי הקוד המופיעים בשאלה. רק דף התשובות יבדק. מחברות הטיוטא יגרסו.

תשובה 1:	תשובה 3:		
תשובה 2:			
4)	1)		
5)	2)		
6)	3)		
תשובה 4 סעיף א:		תשובה 4 סעיף ב:	

לשימוש הבודקים

תשובה 1	תשובה 2	תשובה 3	תשובה 4א	תשובה 4ב	ציון

