

סמסטר א' תשס"ח
בוחן אמצע – 2.3.08

מר שחר גולן
מר טל גרינשפון
גב' דיקלה דותן
דר' מיכל זיו-יוקלסון
פרופ' מייק קודיש

משך הבוחן: שעתיים

חומר עזר - אסור

במבחן זה 4 שאלות, בשווי 25 נקודות כל אחת. ענו על כל השאלות.
השאלות הינן שאלות תכנות. מספר השורות העומדות לרשותכם בדף התשובות רומז על אורך הקוד הנדרש. הקפידו על כתב יד ברור. תשובות מסורבלות או ארוכות מדי לא יזכו בניקוד מלא.
בשאלות בהן לא מצוין אחרת, ניתן לבחור בפתרון רקורסיבי או פתרון שאינו רקורסיבי, לבחירתכם.
שימו לב: החשיבות העליונה היא לנכונות הקוד. מאידך, יעילות, סגנון וחלוקה נכונה לפונקציות עזר חשובים גם הם, ולכן תשובה יעילה ומסוגגנת תזכה בציון גבוה יותר.
בדף האחרון של המבחן ישנה תזכורת למספר פונקציות חשובות, בהן תוכלו להשתמש במהלך המבחן.
אנא רשמו את תשובותיכם **בדפי התשובות בלבד**. המחברת שקיבלתם היא מחברת טיוטה והיא לא תימסר כלל לבדיקה. בסיום הבחינה נשמור אך ורק את דף התשובות. כל שאר החומר יועבר לגריסה. הקפידו לרשום בשני דפי התשובות גם את מספר הנבחן ומספר החדר שבו אתם נבחנים.

שאלה 1 (25 נקודות)

כיתבו בדף התשובות את הפונקציה

```
public static char maxLetter(String str){
    char result;
    // השלימו בדף התשובות
    return result;
}
```

המקבלת מחרוזת כלשהי של תוים (הכוללת גם אותיות A-Z, a-z) ומחזירה את האות שמופיעה מספר רב ביותר של פעמים. אין להבדיל בין אותיות גדולות לקטנות. למשל האות 'A' והאות 'a' נספרות שתייהן כאות 'a'. הפונקציה תחזיר תמיד אות קטנה מתוך האותיות a-z. מומלץ לחלק את הפתרון לפונקציות עזר לבחירתכם.

לדוגמה:

```
System.out.println(maxLetter("aaBBBAA"));
```

P a

```
System.out.println(maxLetter("One. Two. Three. Even. Odd."));
```

P e

הפונקציה מתעלמת מתוים שאינם אותיות. לדוגמה,

```
System.out.println(maxLetter("881771 82&&&6 $$$aabc"));
```

P a

במידה ולא קיימת אות המופיעה יותר פעמים מכל אות אחרת (אין אות "מנצחת"), הפונקציה תחזיר אות כלשהי בעלת מספר רב ביותר של מופעים. לדוגמא: maxLetter("abc") תחזיר או a או b או c. יש להניח שהמחרוזת אינה null ושיש בה לפחות אות אחת.

שאלה 2 (25 נקודות)

מעריך דו מימדי שכל שורותיו בעלות אותו האורך נקרא "טבלה". טבלה של מספרים (שלמים) תקרא ממוינת אם כל שורותיה וכל עמודותיה ממוינות (בסדר לא יורד). השלימו בדף התשובות את הפונקציה

```
public static boolean isSorted(int[] [] table){
    boolean result;
    // השלימו בדף התשובות
    return result;
}
```

המקבלת כפרמטר טבלה table, ומחזירה ערך אמת אם היא ממוינת, וערך שקר אחרת. ניתן להניח כי מערך הקלט table אינו null וכי הוא אכן טבלה חוקית של m שורות ו-n עמודות עבור $n > 0$, $m > 0$. בשאלה זו אין להוסיף פונקציות עזר נוספות.

טבלה לא ממוינת:

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

טבלה ממוינת:

-3	1	4	4
-1	3	5	6
-1	3	7	8

שאלה 3 (25 נקודות)

השלימו בדרך התשובות את הפונקציה הרקורסיבית הבאה, המקבלת מספר שלם לא שלילי ומחזירה את הייצוג הבינארי שלו כמחרוזת. ניתן להניח שהקלט $number \geq 0$. בשאלה זו אין להוסיף פונקציות עזר נוספות.

```
public static String decToBin(int number){
    String answer;
    // השלימו בדרך התשובות
    return answer;
}
```

לדוגמה:

```
System.out.println(decToBin(16));
P 10000
System.out.println(decToBin(6));
P 110
System.out.println(decToBin(1));
P 1
System.out.println(decToBin(0));
P 0
```

שאלה 4 (25 נקודות)

השלימו בדרך התשובות את התוכנית

```
public static boolean splitArray(int[] nums){
    return splitArray(nums, _____);
}
```

```
public static boolean splitArray(int[] nums, _____){
    boolean result;
    // השלימו בדרך התשובות
    return result;
}
```

הפונקציה `splitArray(int[] nums)` מקבלת מערך של מספרים שלמים `nums` ובודקת האם ניתן לחלק את כל איברי המערך לשתי קבוצות כך שסכום המספרים בקבוצה האחת יהיה שווה לסכום המספרים בקבוצה השנייה. כל מספר במערך `nums` יופיע רק באחת מבין שתי הקבוצות: בקבוצה האחת או בקבוצה השנייה. הפונקציה מחזירה ערך אמת (boolean): `true` - במידה וקיימת חלוקה כזו, `false` - אחרת.

עליכם להשתמש בתבנית הנתונה ולהגדיר פונקציה עזר רקורסיבית `splitArray` אשר מקבלת את `nums` ועוד ארגומנטים נוספים.

דוגמאות לתוצאות ריצה:

```
int[] test1 = {2,1,1}, test2 = {2,0,0}, test3 = {2,5,3}, test4 = {-1,5,4};
System.out.println(splitArray(test1)); → true {1,1} ו- {2}
System.out.println(splitArray(test2)); → false
System.out.println(splitArray(test3)); → true {2,3} ו- {5}
System.out.println(splitArray(test4)); → true {-1,5} ו- {4}
```

רמז:

השתמשו בארגומנטים של פונקציה העוזר על מנת לעקוב אחר סכום המספרים בקבוצות או על ההפרש ביניהם).

תזכורת:

- `int length()`
שיטה במחלקה `String` המחזירה את אורך המחרוזת
- `int indexOf(char c)`
שיטה במחלקה `String` המחזירה את האינדקס הראשון במחרוזת, בו מופיע התו `c`,
או -1 אם `c` לא מופיע כלל במחרוזת
- `char charAt(int index)`
שיטה במחלקה `String` המחזירה את התו במיקום `index` במחרוזת
- `String substring(int beginIndex, int endIndex)`
שיטה במחלקה `String`, המחזירה מחרוזת חדשה שהיא תת מחרוזת של עצם המפתח, החל מהמיקום `beginIndex` (כולל) ועד המיקום `endIndex` (לא כולל).
- `double Math.pow(double a, double b)`
פונקציה המחזירה את הערך `a` בחזקת `b`
- `double Math.sqrt(double a)`
פונקציה המחזירה את השורש הריבועי החיובי של `a`
- `double Math.abs(double a)`
פונקציה המחזירה את ערכו המוחלט של `a`
- `int Math.abs(int a)`
פונקציה המחזירה את ערכו המוחלט של `a`