

סמסטר א', 29.12.14

בוחן מבוא לחישוב ב java

<u>מחלקה:</u>	מדעי המחשב ומתמטיקה
<u>מרצה:</u>	גב' אליזבת איצקוביץ
<u>שם הקורס:</u>	מבוא לחישוב
<u>משך הבחינה:</u>	שעה וחצי (90 דקות)

חל איסור על שימוש בכל חומר עזר.

הנחיות כלליות:

- ❖ בבוחן זה 4 שאלות, עליכם לענות על כולן, את הבוחן יש להגיש כקובץ zip או rar בשם **תעודת הזהות** של הסטודנט.
- ❖ המבחן ייבדק בצורה אוטומטית ע"י תוכנת מחשב שתשתמש בשמות המוזכרים להלן.
- ❖ אסור להעתיק, אסור לדבר, אסור לשלוח מידע כלשהו מהמחשב שלכם לכל גורם אחר במהלך הבחינה.
- ❖ גם בחנים לא מלאים יבדקו – אבל בוחן שלא עובר קומפילציה יורדו מספר משמעותי של נקודות.
- ❖ את הפתרון יש להגיש לפני סיום הבחינה – תרגילים שהוגשו באיחור (אפילו של דקה) יקבלו ניקוד חלקי בלבד!

הנחיות לתכנות:

- ❖ יש לבנות Java Project בשם Bohan
- ❖ לפתרון של כל שאלה צריך לבנות **מחלקה נפרדת**.
- ❖ את קובץ הפרויקט ואת השאלון המעודכן ניתן להוריד מ"מודל".
- ❖ ניתן להשתמש בקוד המצורף ובכל אמצעי java.

שאלות הבוחן:

שאלה 1. כתבו פונקציה שמקבלת מספר שלם n , ומחשבת את קבוע מתמטי E , בסיס הלוגריתמים הטבעיים. ערך של מספר זה ניתן למצוא ב- Math.E . הערך המחושב עבור $n=8$ צריך להיות קרוב לערך המתקבל מ- Math.E . נסחה לחישוב קבוע מתמטי E :

$$1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} + \dots + \frac{1}{n!}$$

יש לכתוב מחלקה בשם **Question1**.
בתוך מחלקה זו יש לכתוב פונקציה סטטית המקבלת מספר שלם n ומחזירה ערו של קבוע מתמטי E .

```
public static double calcE(int n) {...}
```

שאלה 2. כתבו פונקציה שמקבלת מחרוזת ומחזירה אמת אם (ורק אם) המחרוזת מייצגת מספר טבעי, כלומר מכילה רק ספרות.

יש לכתוב מחלקה בשם **Question2**.
בתוך מחלקה זו יש לכתוב פונקציה סטטית המקבלת מחרוזת ומחזירה `true` אם המחרוזת מכילה רק ספרות, אחרת הפונקציה מחזירה `false`.

```
public static boolean isInt(String s) {...}
```

דוגמאות:

(1) קלט: "1234", פלט: `true`.

(2) קלט: "34a12", פלט: `false`.

שאלה 3. כתוב פונקציה שמקבלת מערך של מספרים שלמים ומבטלת את כל האיברים הכפילים. הפונקציה מחזירה מערך חדש המכיל את איברי המערך המקורי ללא כפולים.

יש לכתוב מחלקה בשם **Question3**.
בתוך מחלקה זו יש לכתוב פונקציה סטטית המקבלת מערך של שלמים ומחזירה מערך חדש ללא כפילויות:

```
public static int[] delDuplicate(int[] arr) {...}
```

דוגמא:

קלט: $\text{arr} = \{1,2,4,2,6,7,3,2,1\}$, פלט: $\{1,2,4,6,7,3\}$

שאלה 4. כתוב פונקציה שמקבלת מטריצה בת m שורות ו- n עמודות וממיינת את כל שורה שלה בסדר עולה.

יש לכתוב מחלקה בשם **Question4**.

בתוך מחלקה זו יש לכתוב פונקציה סטטית המקבלת מטריצה של שלמים וממיינת את כל שורה שלה בסדר עולה.

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

$$\begin{pmatrix} 1 & 3 & 4 & 5 \\ 2 & 6 & 7 & 8 \\ 0 & 1 & 3 & 9 \end{pmatrix} \text{ פלט: } \begin{pmatrix} 1 & 5 & 4 & 3 \\ 8 & 6 & 7 & 2 \\ 3 & 9 & 1 & 0 \end{pmatrix} \text{ דוגמא: קלט}$$

בהצלחה!

נספח קוד, מצורף לבוחן:

שכולל תוכנית ראשית שמשתמשת במספר פונקציות למיון והדפסה של מערך של שלמים.

```
public class CodeExample {
    public static void main(String[] a) {
        // ***** using the sort (static) function *****
        int[] arr = {1,4,2,6,3,9,0};
        sort(arr);
        printArray(arr);
        System.out.println(round(2.4)+" "+round(2.6));
    }
    /** selection sort: find the smallest entry, put a side ... */
    static void sort (int[] arr) {
        int to = arr.length;
        for(int from=0;from<to-1;from=from+1) {
            int minInd = minIndex(arr,from,to);
            swap(arr, from, minInd);
        }
    }
}
```

```
}
```

```
/** @param: arr : array of integers. returns the minimum index in [from,to). */
```

```
static int minIndex(int[] arr, int from, int to) {
```

```
    int ans = from;
```

```
    for(int i =from+1; i< to ; i=i+1) {
```

```
        if (arr[ans] > arr[i]) ans = i;
```

```
    }
```

```
    return ans;
```

```
}
```

```
/** Note: changes the array - swap i,j values. */
```

```
static void swap(int[] arr, int i, int j) {
```

```
    int tmp = arr[i];
```

```
    arr[i] = arr[j];
```

```
    arr[j] = tmp;
```

```
}
```

```
/** this function simply prints an array (return void). */
```

```
static void printArray(int[] arr) {
```

```
    System.out.println(); // new line
```

```
    for(int i =0; i< arr.length ; i=i+1) {
```

```
        System.out.print(arr[i]+" "); }
```

```
    System.out.println(); // new line
```

```
}
```

```
/** Strings: */
```

- boolean **equals**(String s); // מחזיר אמת אם שתי המחרוזות שוות לוגית
- String **substring**(int start, int end); // [start,end) מחזירה תת-מחרוזת
- int **length**(); // מחזירה את אורך המחרוזת (בתווים)
- char **charAt**(int i) // מחזירה את התו במקום i