

הכנות מונחה עצמים (שפת JAVA)

תרגיל בית אהבה מס' 2. (ארנון ברקת)

ספטמבר, 2020.

Generic, Interface, ArrayList

תאריך חלוקת התרגיל בכיתה: יום שני, 16/12/2019.
התאריך להגשת התרגיל: עד וכולל יום שני, 06/01/2020. (עד השעה 23:45).

יש להגיש אך ורק דרך תפריט המטלות שבאתר הקורס, כפי שהוסבר בתרגול.
שם הקובץ הראשי, שבו יש את `main`, יכיל את שם המגיש ויהיה אך ורק לפי הפורמט הבא: למשל:
עבור תרגיל בית מס' 1: `HW1_AviLevi.java`. עבור תרגיל בית מס' 2: `HW2_AviLevi.java`, וכך הלאה.
שם המחלקה יהיה אף הוא בהתאם. למשל: `public class HW1_AviLevi{`.
בנוסף, הקובץ הראשי יכלול בשורה הראשונה למעלה, הערה ובה השם המלא של המגיש/ה.
אם העבודה מכילה רק קובץ אחד, יש להגיש את קובץ ה-`java` היחיד הזה.
אחרת, יש לזפזף ולהגיש קובץ `zip`, שיכיל רק את קבצי ה-`Java`, ולא פרויקט של `Eclipse` וכו'.
בכל מקרה אין להגיש באתר יותר מקובץ אחד.
על הסטודנט חלה האחריות שהקובץ שהוא הגיש תקין, נשלח בצורה נכונה ומתאים לנדרש.
על התרגילים להיפתח בהצלחה בתוכנת ה-`Eclipse`, ללא שגיאות הידור או אזהרות.
יש לשלב בקוד הערות ותיעוד מתאים. (באנגלית בלבד!)
יש לבצע הזחה של הקוד כנדרש.
יש להקפיד על כללי הנדסת התוכנה ככל הניתן: קוד קצר, לא מסורבל ויעיל הן מבחינת כתיבתו והן מבחינת ריצת התוכנית. לא להשתמש במשתנים סטטיים שלא לצורך, תוכנית כללית שניתנת בקלות לשינויים והרחבות בעתיד, שימוש בקבועים, חלוקה מתאימה לפונקציות, פונקציות עצמאיות (כלומר שאינן תלויות בקוד /משתנה חיצוני), וכיוצ"ב. בנוסף, פתרון התרגיל צריך להיות גם כללי לכל שינוי של הנתונים, שינוי הגדלים של המערכים באם קיימים, וכו'.
כאשר הבדיקה תסתיים, תהיה על כך הודעה באתר. עד שאין את ההודעה, יש עוד עבודות שלא נבדקו.
לא יבדקו תרגילים שמוגשים באיחור ו/או שאינם עומדים בדרישות הנ"ל

בדיקת התרגילים מבוצעת ע"י גיא.

Guy.Kabiri@s.afeka.ac.il

המטלה הנדרשת

כתוב מחלקה *Course* המייצגת קורס שהועבר במכללה בסמסטר הקודם.

יש בו את שני השדות הבאים, ולא יותר :

```
private E idOrName;  
private float avg;
```

מכיוון שלא תמיד זוכרים את מספר הקורס, החליטו לתת אפשרות להזין את זיהוי הקורס או ע"י מספרו, או ע"י שמו.

ולכן, השדה *idOrName*, מייצג או את שם הקורס בטיפוס של *String*, או את מספר הקורס מטיפוס שלם.

בנוסף, כפי שניתן לראות, שומרים גם את הממוצע *avg* של אותו קורס, מטיפוס *float*.

כעת, יש לשאול את המשתמש כמה קורסים יש לו.

ואז לבקש ממנו לבחור לפי מה הוא רוצה הפעם להזין את זיהוי כל הקורסים :

או לפי מספר או לפי שמם. אם יקיש 1 - אז לפי שם. 2 - לפי מספר הקורס.

ניתן להניח שהקלט הוא תמיד תקין ובנוסף הוא תמיד רק 1 או 2, ואין צורך לבדוק או לוודא את זה.

כעת, כתוב מתודה בשם *insert*, ובה יש לקלוט מהמשתמש את הנתונים של הקורסים, כאשר הזנת הנתונים תהיה

כמובן לפי הבחירה שלו. ניתן להניח שהקלט תקין מכל הבחינות, ואין צורך לבדוק או לוודא את זה.

יש להכניס את הקורסים לתוך *ArrayList*.

המתודה תחזיר ל- *main* את ה- *ArrayList*.

כתוב מתודה בשם *print*, ובה יש להדפיס את תוכן ה- *ArrayList* למסך, באופן ישר ומסודר, כמו בדוגמאות

ההרצה.

כעת, כתוב מתודה בשם *fromArrayListToArray*, שתקבל את ה- *ArrayList* הני"ל, תעתיק את הקורסים שיש בו

לתוך מערך 'רגיל' (כלומר כזה עם []), ותחזיר את המערך 'הרגיל' ל- *main*.

כתוב מתודה בשם *print*, שתדפיס את תוכן המערך 'הרגיל'.

במתודה לא יופיע סימן שאלה באף מקום (*wildcard*), וגם לא תופיע המילה *Course*.

בשלב הזה, בצע מיון למערך 'הרגיל' בעזרת פונקציית המיון שיש במחלקה *Arrays*.

המיון יתבצע בסדר עולה לפי השדה *idOrName*, כלומר הוא מכיל שמות אז בסדר עולה לפי א'-ב', ואם הוא מכיל

מספרים שלמים אז בסדר עולה לפי המספרים. יש להדפיס את תוכן המערך לאחר המיון, במתודה שכתבת קודם.

כעת, יש לממש, כלומר לכתוב באופן מלא מתודה למיון של מערך 'רגיל', אבל באופן כזה שהמתודה תהיה גנרית,

כלומר לא יופיע בה בשום מקום הטיפוס של המערך שעליו עושים את המיון.

יש לכתוב את הקוד של המיון, ולצורך כך ניתן להתבסס על שיטת מיון כלשהי שנלמדה, כרצונך, למשל *quickSort*,

מיון בועות וכך הלאה. המיון יתבצע כמו קודם, כלומר בסדר עולה לפי השדה *idOrName*.

לבסוף, הדפס את המערך הממוין, באופן ישר ומסודר למסך, כמו בדוגמאות ההרצה.

אילוצים לתוכנית כולה:

- שמות של מחלקות, מתודות, משתנים ושדות יופיעו באופן המדויק כפי שמופיעים כאן, אסור להוסיף שדות נוספים במחלקה *Course*, ואסור להוסיף מחלקה אחרת.
- המתודות שאותן צריך לכתוב יכתבו מתחת ל- *main*, כלומר לא בתוך המחלקה של ה- *Course*.
- בתוכנית לא תופיע המילה *Object* באף מקום, גם לא פעם אחת, כי לא צריך אותה.
- בתוכנית לא תופיע המילה *instanceOf* וגם לא המילה *getClass*, כלומר יש לעבוד באופן גנרי.
- בתוכנית לא יופיע *warning* (הערה צהובה) באף מקום, אפילו לא פעם אחת, כי יש להתאים את הטיפוס הגנריים כנדרש. (אסור כמובן להשתמש ב- *SuppressWarnings...*)
- ב- *main* לא יופיעו לולאות, כי הלולאות יופיעו במתודות.
- יש לעבוד באופן גנרי ככל הניתן, כלומר למשל לא להשתמש בכלל במילה *Integer.parseInt* (אלא אם זה *Integer.parseInt* בקליטת שלמים מהמשתמש), וכמה שפחות להשתמש במילה *String* וכו'.
- יש לקבל את הפלט לפי דוגמאות ההרצה.

דוגמא מס' 1 להרצת התוכנית:

Please enter how many courses: 5

Please press 1 to insert the courses by names, 2 by id: 1

Please enter name: Java

Please enter avg: 100

Please enter name: Algebra

Please enter avg: 88

Please enter name: Math

Please enter avg: 55

Please enter name: C#

Please enter avg: 99

Please enter name: Singing

Please enter avg: 100

All courses in ArrayList:

```
(      Java,  100.00)
(      Algebra,  88.00)
(      Math,   55.00)
(      C#,    99.00)
(      Singing, 100.00)
```

All courses in 'regular' array:

```
(      Java, 100.00)
(      Algebra, 88.00)
(      Math, 55.00)
(      C#, 99.00)
(      Singing, 100.00)
```

The 'regular' array of courses, after sorting with Arrays.sort:

```
(      Algebra, 88.00)
(      C#, 99.00)
(      Java, 100.00)
(      Math, 55.00)
(      Singing, 100.00)
```

The 'regular' array of courses, after my sorting method:

```
(      Algebra, 88.00)
(      C#, 99.00)
(      Java, 100.00)
(      Math, 55.00)
(      Singing, 100.00)
```

דוגמא מס' 2 להרצת התוכנית:

Please enter how many courses: 5

Please press 1 to insert the courses by names, 2 by id: 2

Please enter id: 9

Please enter avg: 80

Please enter id: 100

Please enter avg: 55

Please enter id: 3

Please enter avg: 99

Please enter id: 888

Please enter avg: 77

Please enter id: 22

Please enter avg: 66

All courses in ArrayList:

```
(      9, 80.00)
(      100, 55.00)
(      3, 99.00)
(      888, 77.00)
```

```
(          22,   66.00)
```

All courses in 'regular' array:

```
(          9,   80.00)
(        100,   55.00)
(          3,   99.00)
(        888,   77.00)
(         22,   66.00)
```

The 'regular' array of courses, after sorting with Arrays.sort:

```
(          3,   99.00)
(          9,   80.00)
(         22,   66.00)
(        100,   55.00)
(        888,   77.00)
```

The 'regular' array of courses, after my sorting method:

```
(          3,   99.00)
(          9,   80.00)
(         22,   66.00)
(        100,   55.00)
(        888,   77.00)
```