

מבני נתונים

מטלה מספר 2

להגשה עד – 13.5.17

הדרכה:

- יש לכתוב קובץ בשם hw2.java, לשים אותו בתוך תיקיה ששמה הוא מספר הזיהוי של התלמיד, לדחוס את התיקיה ולהגיש דרך מודל.
- ניתן להגיש את המטלה באיחור של עד שבוע, ועל כל יום באיחור תרד נקודה אחת מהציון.
- **אזהרה:** על המטלה להיעשות באופן עצמאי בלבד. מותר לדבר עם תלמידים אחרים בכל נושא הקשור למטלה, אבל אסור להסתכל בקוד של תלמיד אחר, ואסור להראות קוד לתלמיד אחר. המטלות עוברות בדיקות אוטומטיות. אם תמצא אפילו שורה אחת זהה בשתי מטלות, שניהם יקבלו ציון 0.

בכתה למדנו אלגוריתם למיון בשם מיון מיזוג Merge Sort. במטלה זו נממש מיון מיזוג למחרוזות עם שימוש בתורים בלבד, ללא שימוש בכל מבנה אחר (גם לא מערך). ניעזר במבנה ArrayBlockingQueue הנמצא בספריות של ג'אווה. לדוגמה, השורה הבאה מצהירה ומאתחלת תור בגודל 10 המאחסן מחרוזות בלבד:

```
ArrayBlockingQueue<String> myQ = new ArrayBlockingQueue<String>(10);
```

הכנסת איבר חדש לראש התור, ומחיקת איבר הנמצא בראש התור:

```
myQ.add("Hello");
```

```
String answer = myQ.poll();
```

```
private static ArrayBlockingQueue<String>
```

```
Merge(ArrayBlockingQueue<String> a, ArrayBlockingQueue<String> b)
```

המקבלת שני תורים עם מחרוזות ממוינות בסדר עולה, ומחזירה תור חדש המאחסן את כל האיברים של a ו- b ממוינים בסדר עולה, בזמן ריצה $O(n)$.

2. ממשו מתודה בשם:

```
public static void MergeSort(ArrayBlockingQueue<String>
```

```
a)
```

הממיינת את האיברים בתור בסדר עולה בזמן ריצה $O(n \log n)$.

שימו לב: במטלה זו לא ניתן להשתמש באף מבנה חוץ מתור, ובפרט לא ניתן להשתמש במערך. כמו כן, ניתן להשתמש רק בפונקציות `add`, `poll`, `offer`, `peek`, `size` של התור, ולא בכל פונקציה אחרת של תור. בהינתן שתי מחרוזות s, t , הפונקציה `s.compare(t)` מחזירה 0 אם הם שווים, מספר שלילי אם s קטן, ומספר חיובי אם s גדול.

בהצלחה רבה