

## תכנות מונחה עצמים מטלה 2 - תכנון ומימוש של גרפים מכוונים וממושקלים ב Java

מטלה זו מוקדשת לתכנון ומימוש מבני נתונים ואלגוריתמים על גרפים (מכוונים וממושקלים). כרגיל, מומלץ להתחיל בתכנון המטלה (ללא קוד) ורק לאחר שיש בידכם תכנון בסיסי כדאי לעבור לשלב מימוש הקוד.

סעיפי המטלה מוגדרים כממשקים בחבילה של [api](#), עליכם לתכנן ולממש שני ממשקים מרכזיים:

1. ממשק של [גרף מכוון ממושקל](#)
2. ממשק של [אלגוריתמים](#) על גרפים (מכוונים וממושקלים).

כאשר בדיקת המטלה תעשה דרך שלוש פונקציות סטטיות ציבוריות שמוגדרות במחלקה הראשית [Ex2.java](#) - שימו לב עליכם להשלים את שלוש הפונקציות הנל - דרכן המטלה שלכם תיבדק. המחלקה גרף מכוון - צריכה לתמוך בטעינה של קובצי json, שימו לב: צירפנו עבורכם מספר [קובצי json של גרפים](#) עליכם לאפשר טעינה של הקבצים הנ"ל, אבל אתם בהחלט יכולים לשמור את הגרפים גם בפורמט אחר - כל עוד אתם תומכים הן בפורמט שלכם והן בפורמט שניתן לכם.

### שלבי עבודה:

1. פתחו פרויקט Github, כתבו ב Readme (באנגלית כמובן). כלל הפיתוח של המטלה צריך להיות ב Github, כל חברי הקבוצה חייבים לתרום לפרויקט.
  2. הוסיפו ל Readme, תכנון של המחלקות העיקריות, הרחיבו על האופן שאתם מתכננים לממש, ולבדוק את המחלקות - לפני שאתם מתחילים לממש.
  3. לאחר שתכננתם את הפתרון שלכם, התחילו במימוש: מומלץ מאוד לשלב בדיקות כבר בשלב הפיתוח (בכל מקרה חובה לכתוב בדיקות JUnit לשיטות הציבוריות, שבדקות הן נכונות והן ביצועים).
  4. ממשו ממשק גרפי שכולל תפריט שמאפשר לטעון גרפים מקבצים, לשמור אותם, לערוך אותם ולהריץ אלגוריתמים עליהם. סעיף זה פתוח לכם מבחינת מימוש ואתם בהחלט יכולים לבחור את אופן המימוש שנראה לכם מתאים ביותר - אבל אם יש קואורדינטות בקודקודים - חובה לאפשר הצגה גרפית שלהם בקואורדינטות יחסיות (במילים אחרות - תוודאו שאתם מצליחים להציג את הגרפים שסופקו לכם).
- הקפידו לייצר קובץ jar (ראו הסבר, ואחד נוסף) שמקבל את שם קובץ ה json של הגרף - ומריץ את ה GUI -**  
**הקפידו לקרוא לקובץ Ex2.jar כך שניתן יהיה להריץ את הקובץ משורת הפקדה באופן הבא:**  
**java -jar Ex2.jar G1.json**
5. לאחר שתשלימו את כתיבת התוכנית, השלימו ב Readme את הדברים הבאים:
    - ניתוח של הביצועים של האלגוריתמים שלכם על גרפים בגודל 1000, 10000, 100000, 1000000 (בעלי דרגת צלעות ממוצעת של 20 צלעות לקודקוד: נכנסות + יוצאות - סה"כ 20).
    - הסבר על אופן ההורדה, ההרצה והשימוש בממשק הגרפי.

את המטלה יש להגיש עד ה **13.12.2021** כקישור יחיד ל Github דרך הפורום ההגשה, כל הקישורים נמצאים בקישור [הבא](#):

בהצלחה!!

## Q & A

1. Can I use javaFX (instead of java swing)?

Ans: Sure, just

2. Please elaborate on the TSP method:

Ans: The TSP method works on a directed graph  $G<V,E>$ , it gets a set (say S) of nodes (S is a subset of V). The TSP method computes a path (say P) where P is an ordered **collection** of connected nodes.

Note: P needs to contain S, and each consecutive ordered pair of nodes in P is a directed edge in E). The objective function is to minimize the sum of the weights in the edges in P.

Note: P does NOT need to be a set! It may contain multiple appearances of the same node, **yet the pairs of consecutive nodes in P is a SET of directed edges.**

3. Can I modify the interfaces?

Ans: Not at all, the interfaces should remain "untouched".

4. Do I need to implement all the methods in the [java.util.Iterator](#)? (including the default methods)

Ans: in short, Yes, all the api of the iterator should work! Make sure the "remove" and the "forEachRemaining" methods work well, e.g.,:

```
List<Integer> input = Arrays.asList(5, 3, 11, 15, 9, 2, 5, 11);  
Iterator<Integer> itr = input.iterator();  
itr.forEachRemaining(num -> System.out.print(num+" "));
```

see: this [link](#) for more examples.

Note: the order of the elements as traversed by the Iterator is not given - any order is acceptable.

5. Where is the Google sheet with all the Google form data?

Ans: due to few cases of "wrong-doing-fork & copy" - the file was hidden.

6. Can we get few more days to complete Ex2 properly?

Ans: OK, you have till Monday 13/12/2021 (at 23:59)

- 7.