תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

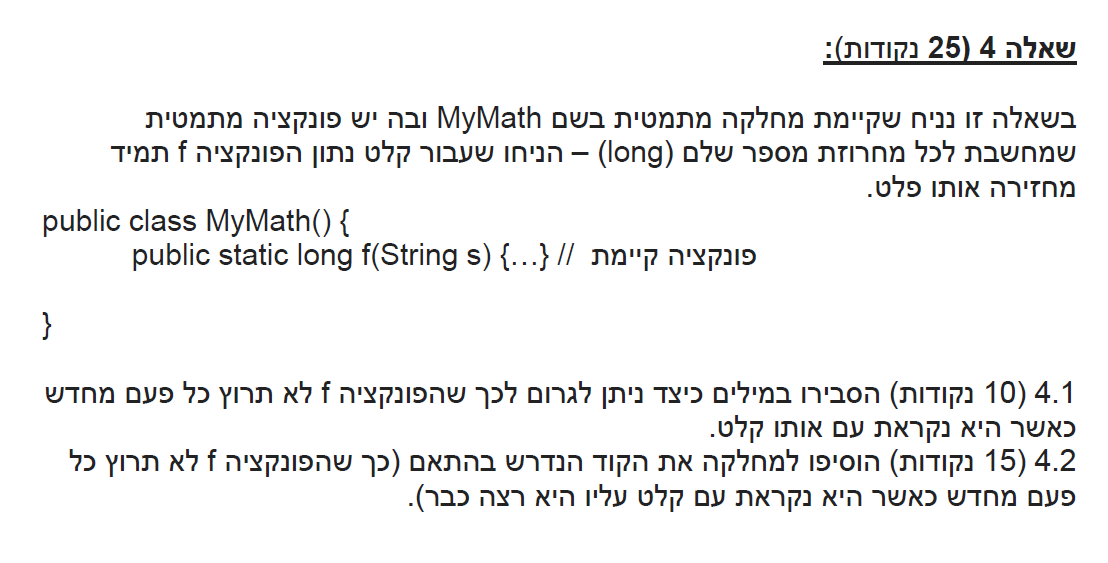
נניח ונוכל להשתמש ב- תהליכונים, נחלק את הרשימה ל רשימות בערך באותו גודל.

ניצור תהליכונים ונמפה לכל תהליכון שיפעיל merge sort על תת רשימה כלשהי.

נחכה שכלל התהליכונים יסיימו.

לאחר שכלל התהליכונים יסיימו, נאחד כל שתי רשימות לרשימה אחת ונמיין אותה, נמשיך כך עד שברשימה המאוחדת יהיו כל האיברים מהרשימה המקורית.

**הערה:** בחרתי לפתור את זה עם merge sort כדי שאוכל להשתמש בפונקציה merge כדי להמשיך ולמזג את תתי הרשימות, אך אפשר לבחור כל מיון אחר ולהתאים אותו לצורך.



**סעיף 4.1**

נוסיף למחלקה שדה של HashMap<String, Long> ובכל הפעלה של הפונקציה נבדוק אם הקלט s הוא מפתח בהאשמאפ, אם כן, נחזיר את הערך שלו, אחרת נבצע את החישוב ונשמור בהאשמאפ את s כמפתח ואת התוצאה של החישוב כפלט.

**סעיף 4.2**

Private static HashMap<String, Long> map = new HashMap<>();

בתחילת הפונקציה f

If(MyMath.map.containsKey(s){

Return MyMath.map.get(s);

}

//rest of the function

בסוף הפונקציה

MyMath.map.put(s, out);

return out;

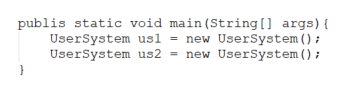
כאשר out הוא התוצאה של החישוב.

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

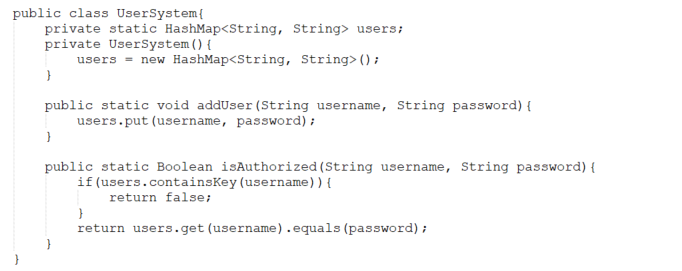
**סעיף 1:**

הבעיה התכנונית נובעת בתיעוד בראש המחלקה ובאופן המימוש. במימוש נדרש שיהיה אך ורק משתנה אחד אך מאחר והconstructor הוא public, ניתן ליצור כמה אובייקטים כאלה שרוצים.



**סעיף 2:**

נתאים את המחלקה להיות מחלקת סינגלטון.



תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

1. מטרת הקוד היא לשמור בתוך סט של מחרוזות את כל המחרוזות המייצגות מספרים מ-0 עד מיליון.  
   בסוף אמור להיות מודפס Thread map size=1000000
2. מה שבפועל יקרה הוא מאחר ושני התרדים משתמשים במשתנה s, מה שיקרה זה שהם בכל פעם יפריעו אחד לשני, יווצר race condition בין התרדים וכתוצאה מכך ככל הנראה לא יכנסו לסט כל המחרוזות וגודל הסט לא יהיה 1000000
3. בהגדרת התרד הראשונה נחליף את שם המשתנה של המחרוזת להיות s1 ובתוך הפונקציה run נתאים את השמות להגדרה החדשה.  
   מה שיקרה זה שלא תהיה תחרות על המשתנים, התרדים ירוצו בנפרד לגמרי מבלי לנסות לגשת לתרד משותף ובכך לא יפריעו אחד לשני.  
   אופציה נוספת לפתרון הבעיה הוא שימוש בסנכרון עבור המשתנה המשותף s



1. הבעיה העקרונית היא שבמידה ונרצה להוסיף סוג חדש של צורה נצטרך להוסיף התנייה נוספת על מנת לטפל בה כמו שצריך, מצב זה סותר את עקרון Open/Close שמ-SOLID.
2. נשאיר במחלקה רק את המשתנה size ונהפוך את print לפונקציה אבסטרקטית, ניצור 3 מחלקות חדשות שכולן יורשות מShape כאשר כל אחת מהן תדרוס את print.  
   נהפוך את כל המחלקה לאבסטרקטית.
3. תמונה שמכילה טקסט

   התיאור נוצר באופן אוטומטי