

פיתוח תוכנה מתקדם 2 – סמסטר ב' מועד ב' תשפ"ב

תזכורת: כתובת מערכת הבדיקות: <https://cktest.cs.colman.ac.il/>. שם הקורס PTM2, מועד ב'. לאחר הורדת המבחן ממערכת הבדיקות. העתיקו את כל קובצי ה Java לתוך הפרויקט ב package בשם **test**. במבחן זה 3 שאלות, חובה לענות על כל 3 השאלות ולהגיש למערכת הבדיקות במוד הגשה סופית לפני סוף המבחן.

שאלה 1 – תכנות מקבילי באמצעות ת'רדים (30 נק')

בקובץ Student.java נתונה המחלקה Student. הבנאי שלה סופר כמה סטודנטים נוצרו בתוכנית באמצעות המשתנה הסטטי count. אולם, בבדיקה אנו מייצרים מס' אקראי וגדול של סטודנטים באמצעות 10 ת'רדים שונים ולכן עלולים לקבל מספר סטודנטים שונה מהצפוי. במקרה כזה, בהתאם להפרש במספר הסטודנטים יורד החלק היחסי מ 30 הנקודות של השאלה.

עליכם לתקן את הקוד ב Student כך שהספירה של הסטודנטים תהיה Thread safe. אולם אסור שימוש כלשהו במילים השמורות synchronized או wait! (הגבלה זו תיבדק ידנית לאחר הבחינה).

שאלה זו נבדקת אוטומטית. הקפידו להגיש קוד שמתקפל ורץ ללא שגיאות ריצה. עם זאת, שגיאות קומפילציה או ריצה שמונעות בדיקה אוטומטית יגוררו קנס של 10 נק' ובדיקה ידנית.

שאלה 2 - תכנות מקבילי באמצעות ת'רדים (35 נק')

בקובץ Actor.java נתונה לכם המחלקה האבסטרקטית Actor. היא אינה לעריכה או להגשה. מחלקה זו מגדירה לכם:

- בנאי שמקבל ID מסוג int
- מתודה אבסטרקטית addMessage המקבלת מחרוזת
- מתודה אבסטרקטית close

עליכם לממש את המחלקה GenericActiveActor ע"פ ההגדרות הבאות:

- היא סוג של Actor
- בבנאי, מלבד ID, היא תקבל `Consumer<String>`
- היא תפעל כ Active Object
 - שמכיל ת'רד אחד בלבד!
 - שכל עוד לא הופעלה המתודה `close()`
 - הוא שולף מתוך תור שהוא thread safe הודעות מסוג String
 - אך אם התור ריק הוא "נח"
 - ומפעיל על הודעות אלו את ה `Consumer`.
- במימוש ל `addMessage` כמובן נכניס את המחרוזת לתור
- במימוש למתודה `close` נגרום לכך שכל ההודעות שנתקבלו לפני הפעלת המתודה יעובדו כרגיל ע"פ הסדר, אך לאחר מכן פעולתנו תיפסק והת'רד ייסגר.

בבדיקה אנו מייצרים 3 Actors שונים כל אחד עם לוגיקה משלו המתקשרים זה עם זה. אנו בודקים שכל אחד מהם יצר בדיוק ת'רד אחד, וכן שאלו נסגרו בסוף התוכנית. כמובן נבדקת גם הפעלתם. מוד האימון זהה למוד הגשה. שאלה זו תיבדק **באופן אוטומטי בלבד**. חובה שהקוד יעבור קומפילציה וירוך ללא שגיאות.

שאלה 3 - אופטימיזציות קוד (35 נק')

בהינתן רשימה של נקודות (אובייקטי Point2D) אנו רוצים למצוא את המרחק המקסימלי הקיים בין 2 נקודות כלשהן בתוך הרשימה. בקובץ BadCode.java מצוי מימוש לא יעיל. עליכם לערוך או לממש מחדש את הקוד בקובץ GoodCode.java כך שהאימפלמנטציה תהיה לפחות פי 5 יותר מהירה.

כמובן, יש להחזיר את התוצאה הנכונה.

מוד האימון **זהה** למוד ההגשה והוא ניתן בקובץ MainTrain3.java.

תחילה מיצרים קלט של מערך אקראי.

לאחר מכן מתבצעת מדידת זמן של BadCode ושל המימוש שלכם ב GoodCode.

קוד שהוא לפחות פי 5 יותר מהיר יקבל את מלוא 35 הנק'

קוד שהוא פחות מפי 5 יותר מהיר יקבל את החלק היחסי.

שאלה זו אף תיבדק ידנית. עם זאת הקנס על קוד עם שגיאת ריצה או קומפילציה הוא 5 נק'.

הגשה

עליכם להיכנס למערכת הבדיקות בכתובת: <https://cktest.cs.colman.ac.il/> ולהגיש ל PTM2 ומועד ב' את הקבצים Student.java, GenericActiveActor.java, GoodCode.java ,

בכל הגשה יש להגיש את כל הקבצים (ולהתייחס לפלט רק של השאלות שעניתם עליהן)

ניתן להגיש במוד אימון ובמוד הגשה כמה פעמים שתמצאו עד לסוף המבחן.

בסוף המבחן יש להגיש **במוד הגשה ואז במוד הגשה סופית**. אחריה תקבלו מס' אסמכתא בין 4 ספרות. לאחר הגשה במוד זה לא תוכלו להגיש יותר.

בהצלחה!