**Python Multithreading**

**תרגיל 1**

כתבו פונקציה המבצעת כפל מטריצות התומך בmultithreading.  
המטריצות לא בהכרח יהיו ריבועיות.  
במידה ולא ניתן לבצע את ההכפלה זרקו חריגה.  
מספר התרדים המקסימלי יהיה מספר העמודות במטריצה הראשונה בקלט.

**תרגיל 2**

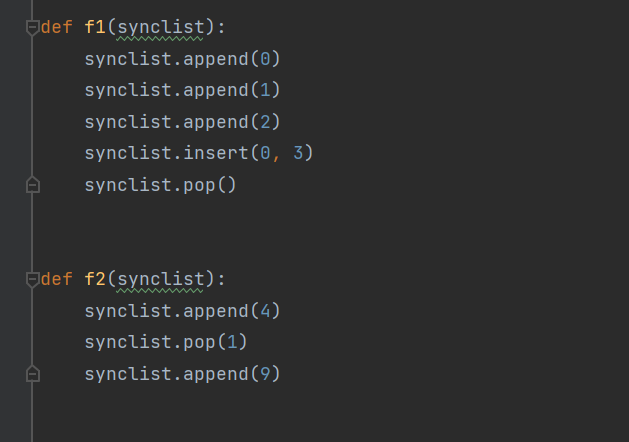
כתבו מחלקה של רשימה סינכרונית בפייתון.  
על הרשימה לתמוך בפונקציות הבאות על בסיס הפונקציות של מבני הנתונים list בפייתון:

* הכנסה לסוף הרשימה
* הכנסה לתא ספציפי
* מחיקת איבר אחרון
* מחיקת איבר ספציפי
* גישה לתא כלשהו ברשימה
* מחיקת כל האיברים
* הוספת איברי רשימה סינכרונית אחרת לסוף הרשימה הסינכרונית
* \_\_str\_\_ על מנת שיהיה ניתן להדפיס את הרשימה הסינכרונית

לבסוף צרו 2 פונקציות המתארות סימולציות של 2 תרדים(אחת לכל תרד) והריצו אותן

עליכם לדאוג שבשתי הפונקציות האלה תופיע כל פונקציה שכתבתם למעלה לפחות פעם אחת

לדוגמה:



הריצו מיין פשוט ובדקו האם כלל הפונקציות שלכם תקינות

**תרגיל 3**

כתבו מחלקה Node המייצגת קודקוד בגרף מכוון המכילה את השדות הבאים:

אינדקס הקודקוד, דרגת כניסה, דרגה יציאה(הדרגות בשלב הזה יהיו תמיד 0), צבע הקודקוד(לבן/אפור/שחור או לפי 1/0/1- לבחירתכם), משקל הקודקוד, ערך X, ערך Y(על מנת שנוכל להציגו במערכת צירים קרטזית)

הוסיפו למחלקה גטרים, סטרים ופונקציות כרצונכם.

כתבו פונקציה המקבלת פרמטר המייצג מספר קודקודים רצוי ומחזירה רשימה של קודקודים אקראיים, שימו לב שאינדקס הקודקוד ייחודי לכל קודקוד, יתר הפרמטרים יכולים להיות אקראיים.

הוסיפו פונקציה המקבלת Node שומרת אותו לקובץ json

צרו מיין ובו צרו רשימה של 10 קודקודים אקראיים ותשמרו אותם בתור קובץ json

הוסיפו פונקציה שבה כל הקודקודים ישמרו באותו הקובץ