

## מטלת פייתון – שיפור ביצועים

משקל המטלה 10% מהציון הסופי.

לפני תחילת המטלה, אנא ודאו שהמימוש שלכם נכון: מימוש נכון חשוב יותר ממימוש יעיל.

- אם קיבלתם ממני הערות לשיפור על המימוש, בעל-פה או בכתב, אנא תקנו ושילחו לי לאישור לפני שתתחילו לעבוד על המטלה.
- אם קיבלתם ציון לא מלא על מטלת המימוש ואתם לא מבינים למה, או שקיבלתם ציון מלא אבל אתם בכל-זאת לא בטוחים שהמימוש שלכם נכון – בואו להתייעץ איתי בשעת הקבלה הקרובה, לפני תחילת המימוש.

### א. השוואת ביצועים

בחלק זה יש לבדוק את ביצועי האלגוריתמים במאמר שלכם על קלטים אקראיים גדולים, ולהשוות אותם לאלגוריתמים אחרים. יש להשתמש ב- `experiments-csv` להגדרת הניסוי, הרצתו, וציור גרפים של התוצאות, כפי שהוסבר בהרצאה.

**מה להשוות בדיוק?** – זה תלוי בבעיה שהאלגוריתם בא לפתור, לדוגמה:

- אם מדובר באלגוריתם לחלוקה הוגנת, אז אפשר להשוות את סכום ערכי השחקנים, ואת הערך המינימלי של שחקן.
- אם מדובר באלגוריתם למציאת שידוך גדול ביותר, או חלוקה קטנה ביותר – אז אפשר להשוות את גודל השידוך או את גודל החלוקה.
- בנוסף, יש להשוות גם את זמן הריצה.

### לאיזה אלגוריתמים להשוות?

- אם במאמר שלכם יש כמה אלגוריתמים, אז תשוו אותם אחד לשני.
- אם יש אלגוריתם אחד עם כמה פרמטרים, אז תשוו הרצות שונות של האלגוריתם עם פרמטרים שונים.
- אם האלגוריתם נמצא בספריה שיש בה אלגוריתמים נוספים הפותרים אותה בעיה, אז יש להשוות לאלגוריתמים מאותה ספריה.
- אם אין עוד אלגוריתמים דומים, אז תשוו לאלגוריתם היוריסטי פשוט כלשהו, הפותר את הבעיה בקירוב.

### על איזה קלטים להשוות?

- יש להשוות על קלטים אקראיים בגודל עולה.
- אפשר להעלות את הגודל עד שזמן הריצה מגיע לכ- 30-60 שניות, בהתאם לאלגוריתם.
- כדי לחשב את מגבלת הזמן באופן אוטומטי, ניתן להשתמש בהרצה עם מגבלת זמן כפי שנלמד בהרצאה.

### איך להגיש?

- יש לשים את הקוד והתוצאות בתיקיה נפרדת בגיטהאב (לא בתיקיה של הקוד), לדוגמה בתיקיה `simulations` או `examples`.



## ב. שיפור הביצועים

יש לבצע חלק זה בענף (branch) נפרד בגיט, כדי שלא יסבך את הקוד בענף הראשי.

השתמשו באחת הדרכים שנלמדו בהרצאות (זכרון-מטמון, דגמי עיצוב, ריבוי תהליכים, שילוב C או C++) כדי להאיץ את זמן הריצה של האלגוריתם שלכם. בחרו את השיפור הטוב ביותר לקוד שלכם, בהתאם לאופי הקוד.

- עצה: כדי למצוא שיפורים טובים, הריצו את האלגוריתם על קלט גדול, ובדקו בעזרת הלוגים איפה המעבד מבלה את רוב הזמן. הכי טוב למצוא קטע קוד קצר, שרץ הרבה פעמים (למשל בלולאה פנימית), ולהתמקד בו.
- אם לא הצלחתם למצוא שיפור טוב בדרכים שנלמדו בהרצאה, נסו למצוא שיפור ע"י שינוי פרמטרים כלשהם של האלגוריתם.

בצעו שוב את סעיף א לאחר השיפור: ציירו גרף המשווה את זמן הריצה כפונקציה של גודל הקלט לפני ואחרי השיפור, ובדקו מהו הקלט הגדול ביותר שהאלגוריתם יכול לעבד ב-60 שניות.

## הגשה

ההגשה אישית. סטודנטים שכתבו את המימוש בצוותים, יבצעו שיפורים שונים לאותו קוד.

