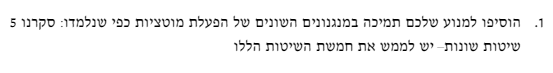
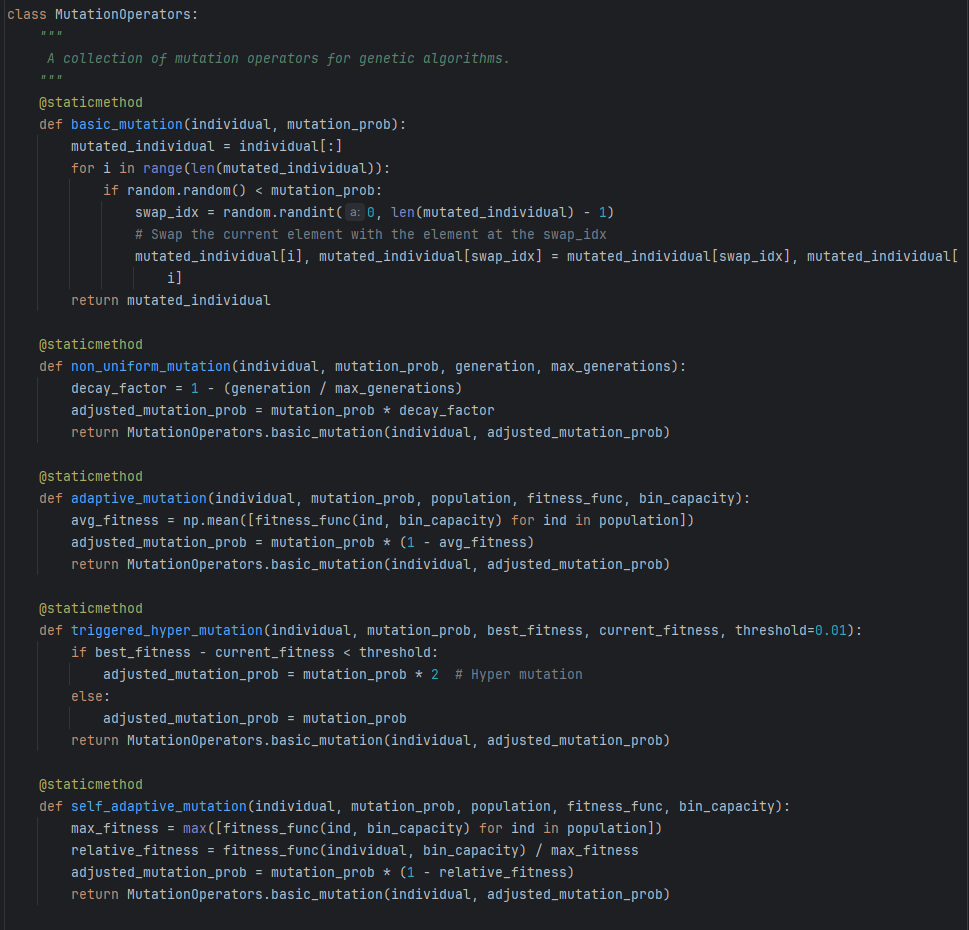
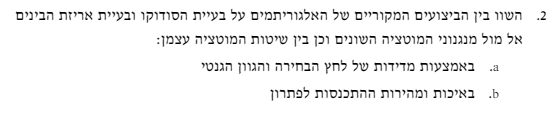
**דוח מעבדה משימה 2**

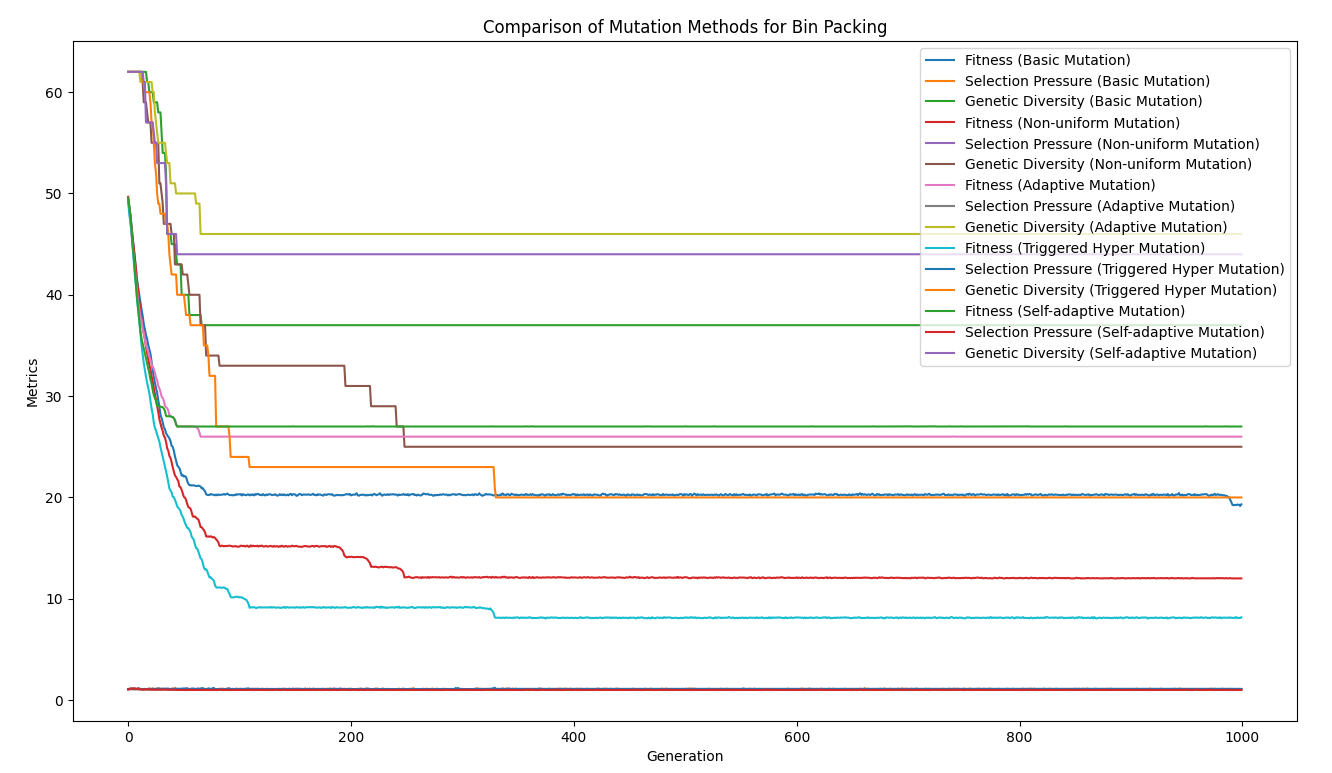
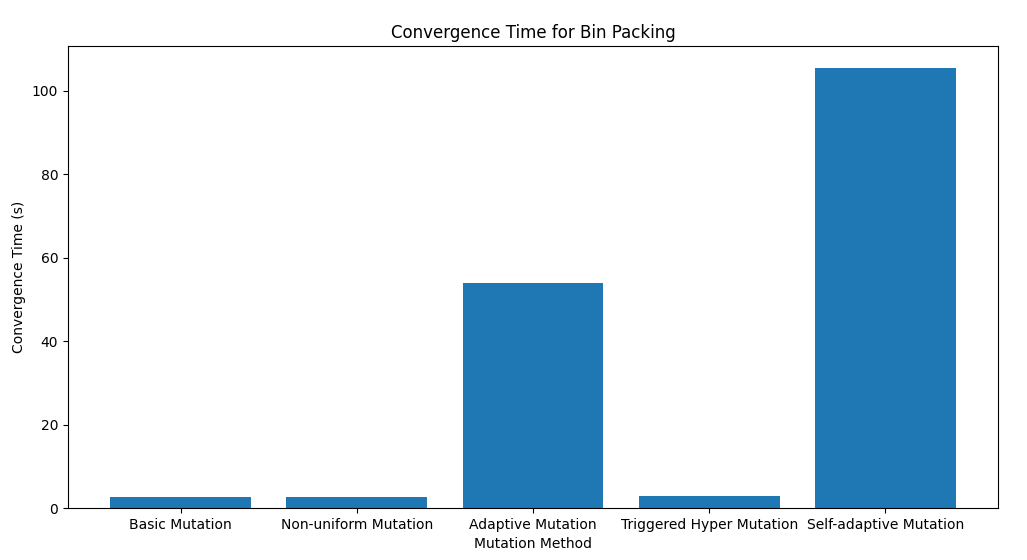
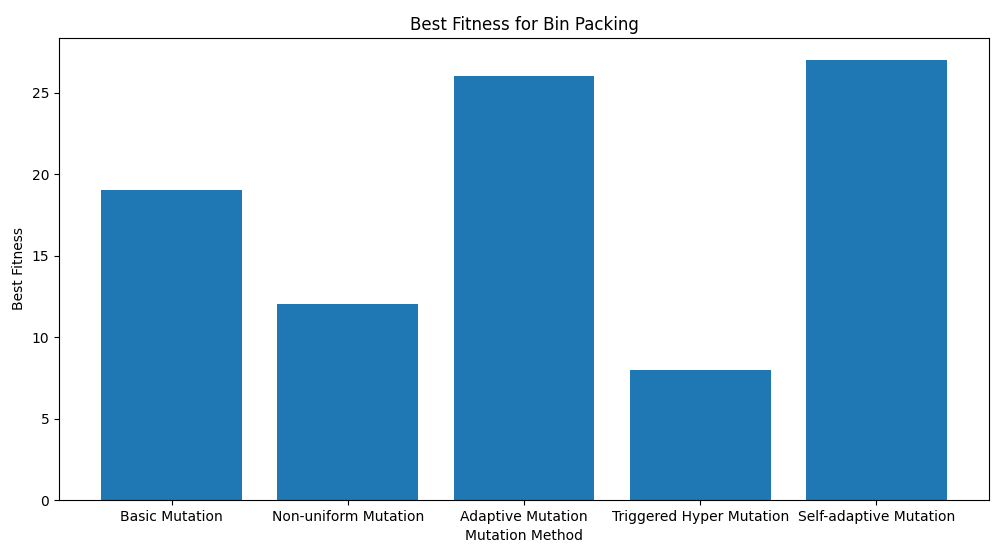


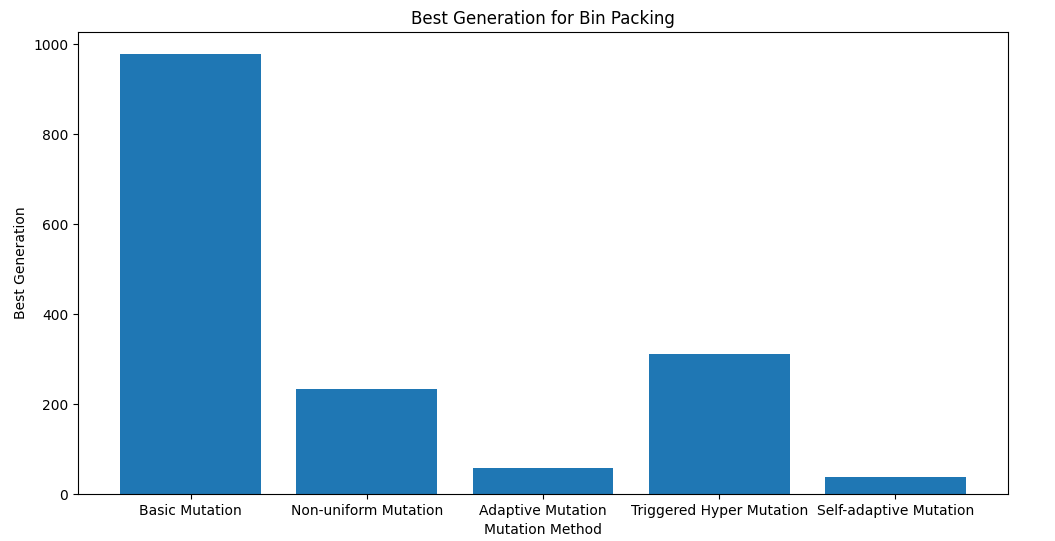
עבור סעיף זה נכתבו הפונקציות הבאות:

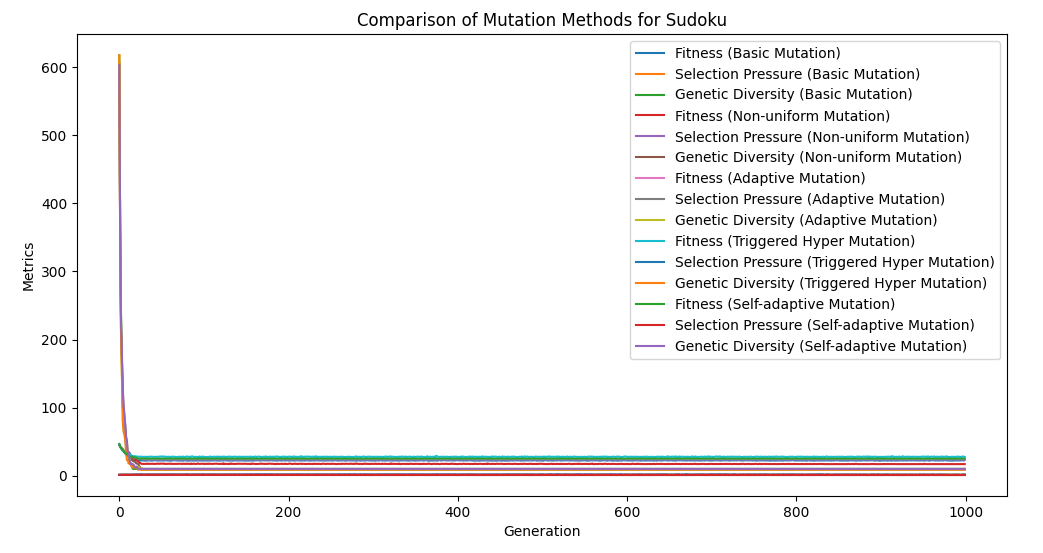


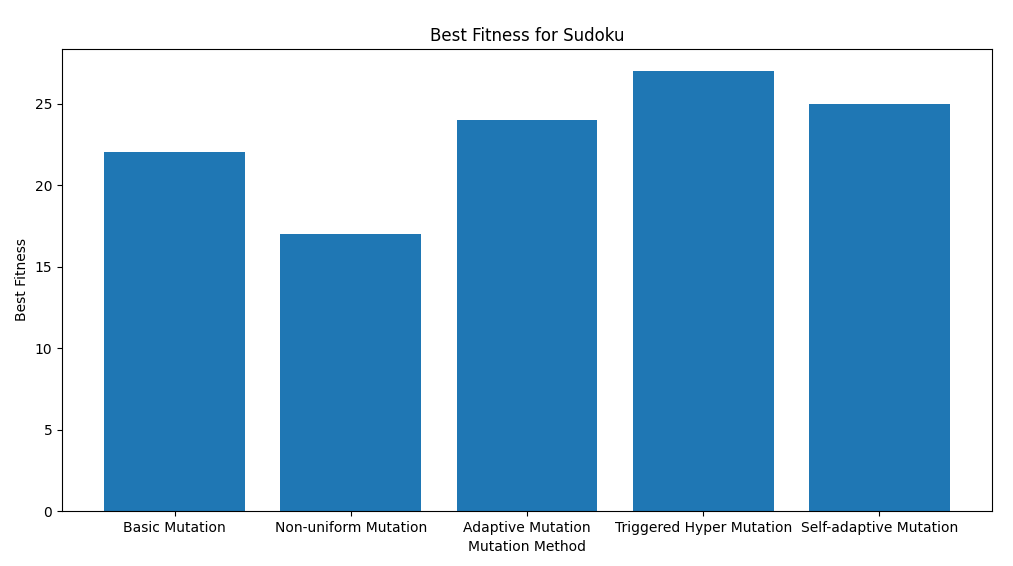


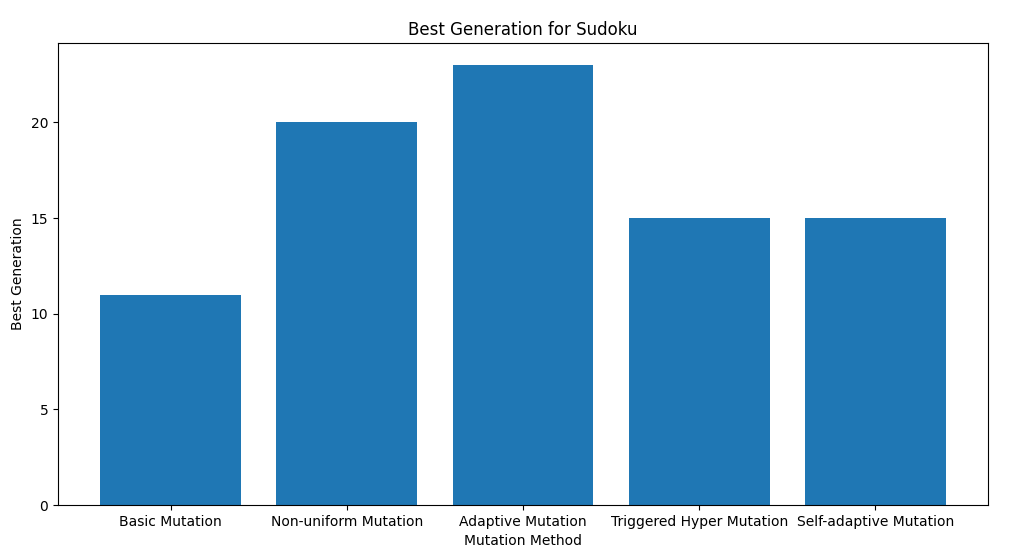
על מנת לענות על סעיף זה יצרנו סימולציה שהשוואתה את הנדרש בשני הסעיפים והגרפים המצורפים הינם ההשוואות הנדרשות. (סימולציה לא צורפה למטלה)

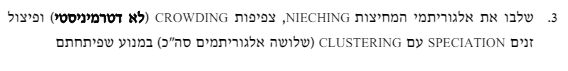
  



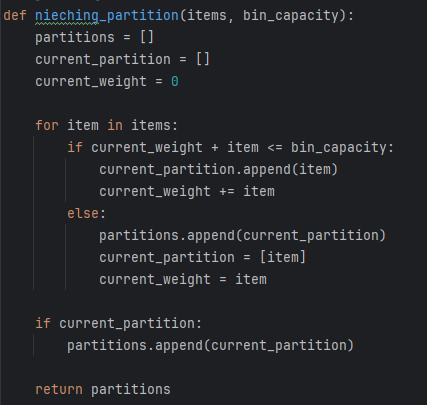


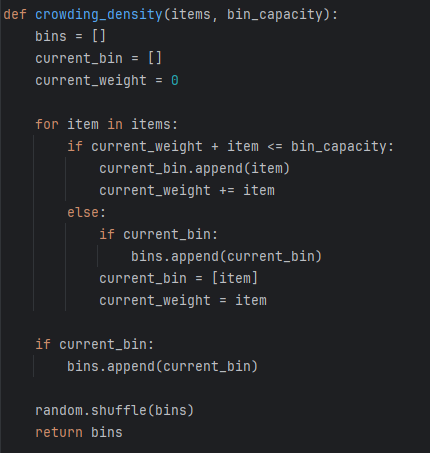


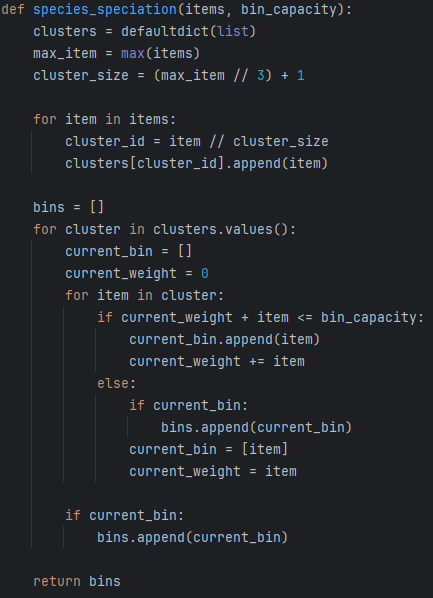


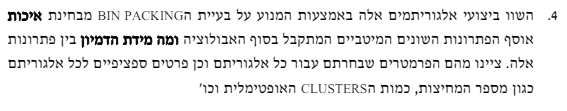


עבור סעיף זה נכתבו הפונקציות הבאות:









הפרמטרים שהוכנסו היו לשלושתם הם:

גודל האוכלוסייה -> 100

דורות -> 1000

קצב מוטציה -> 0.01

מספר השקים/תאים -> 150

עבור – Nieching Partition

מספר המחציות נוצרים באופן דינמי על בסיס סכום המשקלים כדי לא לעבור את המקסימום שניתן לשים,

פונקציית הFitness דואגת לכך.

עבור – Crowed density

מספר התאים ממולא עד למקסימום ואז אנחנו עושים Shuffle באופן רנדומלי כדי לסמלץ Crowding.

פונקציית הfitness דואגת לספור את מספר התאים שבשימוש לאחר הshuffle.

עבור – Species speciation

הקלסטרים נוצרים על בסיס גודל ה-item כאשר קלסטר הוא שליש מהמשקל המקסימלי של ה-item, פונקציית הfitness סופרת את מספר התאים בשימוש לאחר הקיבוץ של הפריטים למינים Species.

מבחינת התוצאות שהתקבלו:

לאחר לא מעט ריצות ניתן לומר שלפעמים מספר הדורות למציאת הפתרון היה דומה אך מספר התאים שנדרש היה שונה, לפעמים מספר התאים בשימוש היה דומה אך מספר הדורות היה שונה.

עיקר הדמיון ניתן למצוא בין שניה האלגוריתמים הראשונים.

איכות הפתרון, לרוב האלגוריתם עם הפתרונות הטובים ביותר היה Nieching Partition כאשר אחריו היה Crowed density, הכי פחות איכותי היה Species speciation.

מחינת כמו הדורות ניתן להסיק כי שלושת האלגוריתמים יחסית מאוזנים.

ניתן לסכם את תוצאות הסימולציה במונחים של מספר התאים בשימוש ומספר הדורות הנדרשים כדי להגיע לפתרון האופטימלי.

