20/12/16

ממן12 אלגוריתם גנטי

מגיש: דב האוזר, 27933092

1. **שאלה 1**
2. שאלה 7 בעמוד 133

ב. שאלה 19 בעמוד 135

בשיטת הנישות, יש סיכוי גדול יותר שכל תת אוכלוסיה (נישה) תתכנס למינימום מקומי אחר. ולכן המיזוגים ביניהם יוכלו להסיט כל תת אוכלוסיה מהמינימום המקומי שאליו היא נמשכת/מתכנסת.

כלומר, בזמן המיזוג לכל תת אוכלוסיה יכנסו פרטים מתתי אוכלוסיות אחרות שיכולים ל"התחרות" עם

המינימום המקומי שנמצא בתת האוכלוסייה ומנצח את כל שאר הפריטים בה.

1. **שאלה 2**

סעיף 1

הקובץ zip כולל

1. את המסמך הזה
2. קובץ ריצה בשם mmn12\_ga.exe, שהוא אפליקציית GUI.
3. ספיריה בשםonly source שכוללת את קבצי הקוד בלבד
4. ספיריה בשם project, שכוללת את קבצי הקוד וגם ופרויקט ריצה.

הפרויקט נכתב ב #C על Visual studio 2013 וקובץ הפרוייקט הוא Application.sln

הפרמטרים הם:

Refresh rate: קצב עדכון הגרף בשניות

Population size: גדול האוכלוסיה

Crossover probability: ההסתברות לשחלוף

Mutate Probability: ההסתברות למוטציה

Selection range: אחוז סינון האוכלוסייה לפני הבחירה. כלומר, לפני הבחירה אני בוחר

אחוז מסוים מהטובים ביותר ומהם עושה את הבחירה הרנדומלית.

הייתי צריך את זה בזמן הפיתוח, אחרי זה ראיתי שזה רץ גם מספיק

טוב בלי זה.

Local minimum detection period:

מדד לזיהוי מינימום לוקלי (בדורות). כלומר, אם האוכלוסייה לא

משתפרת x דורות אני מזהה מינימום לוקלי ומנסה לברוח ממנו.

Elits: מספר אליטות של פריטים

Instant minimum fitness: (לא ניתן לעריכה). ערך הקשירות המינימלי באוכלוסיה

Minimum fitness ever: (לא ניתן לעריכה). ערך הקשירות המינימלי באוכלוסיה מתחילת הריצה

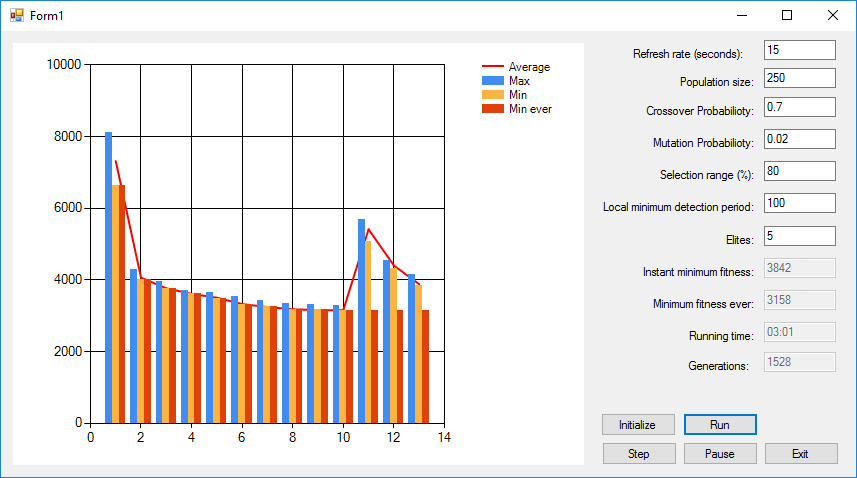
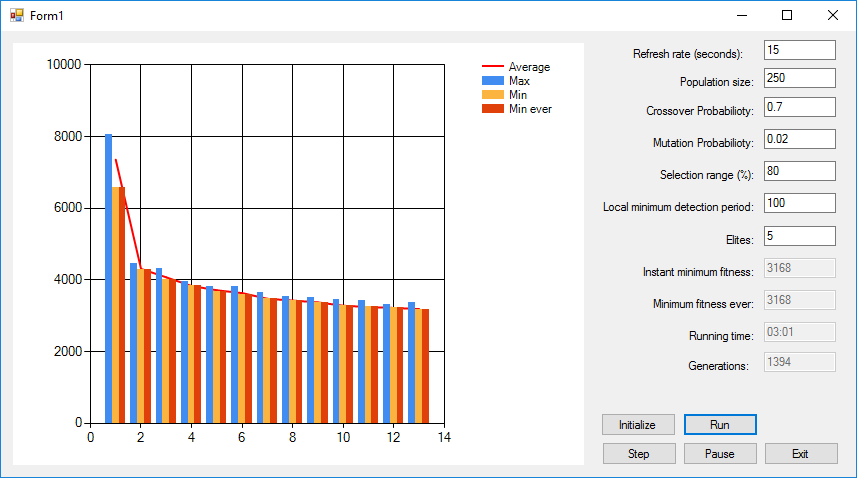
Running time: (לא ניתן לעריכה). זמן ריצה

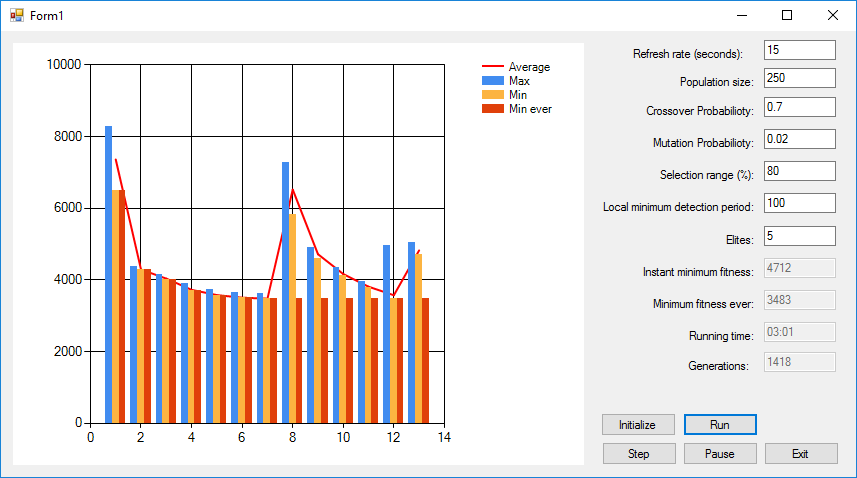
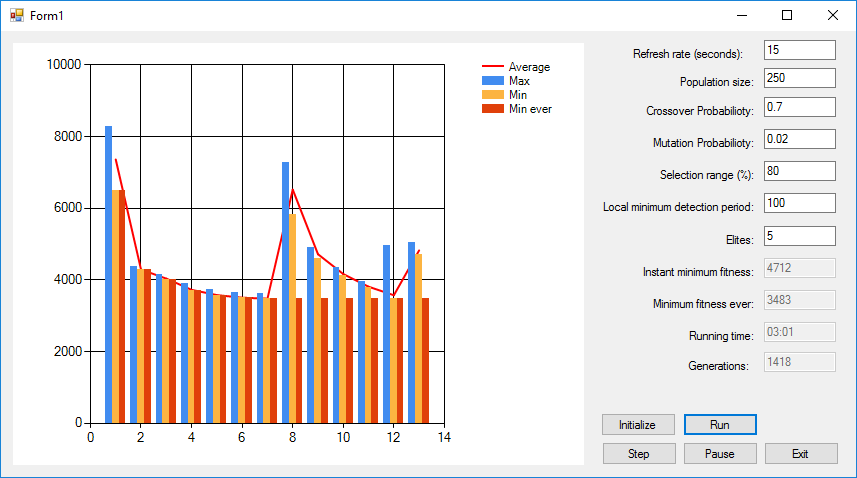
Generations: (לא ניתן לעריכה). מספר דורות מתחילת הריצה

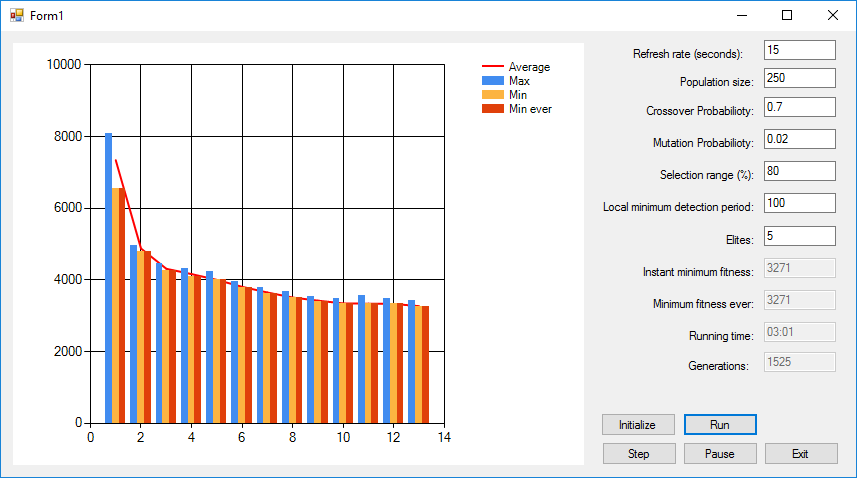
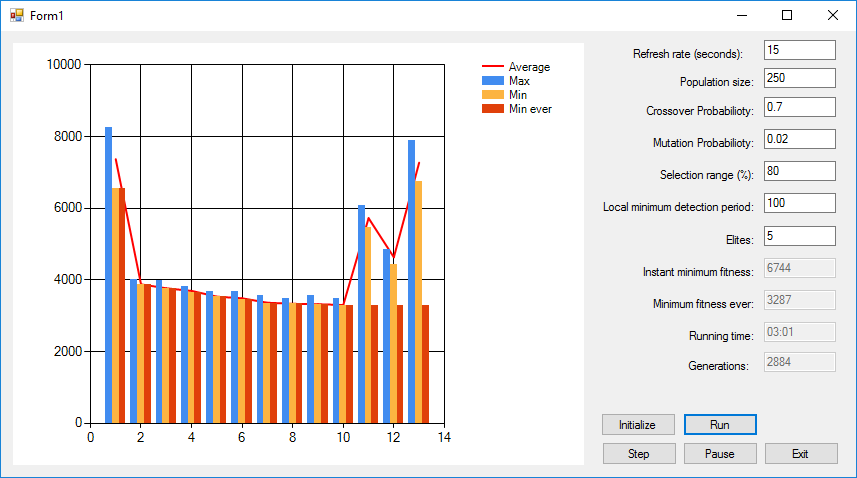
סעיף 2

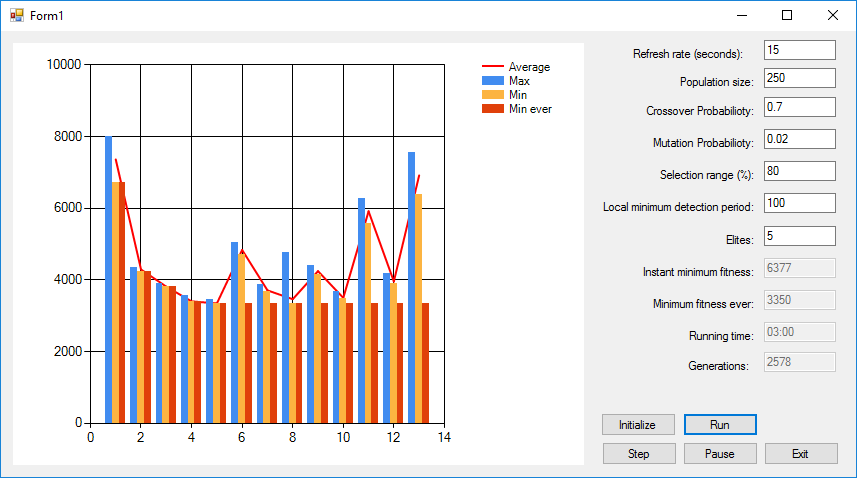
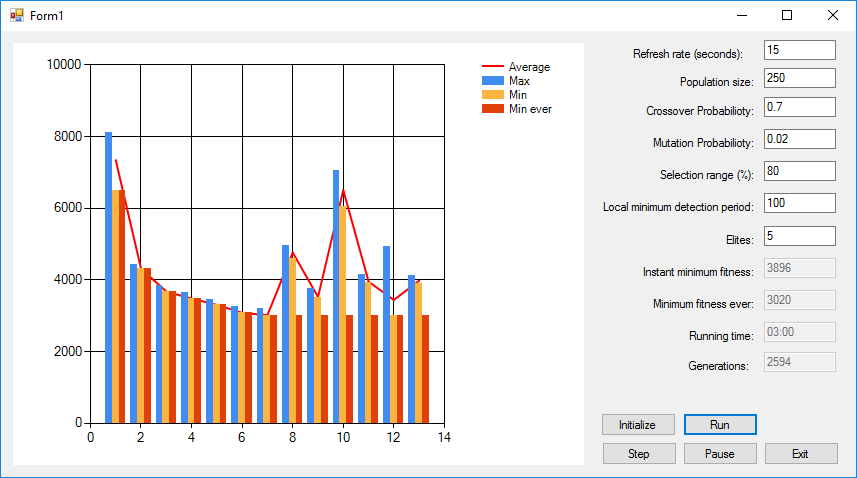
הגרפים הם עם עדכון כל 15 שניות ולא 30, כי היה לי יותר נוח לראות את זה ככה.

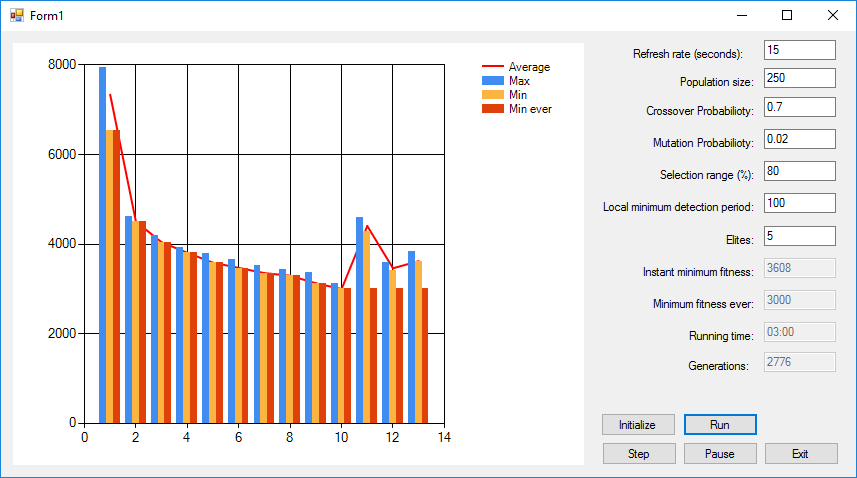
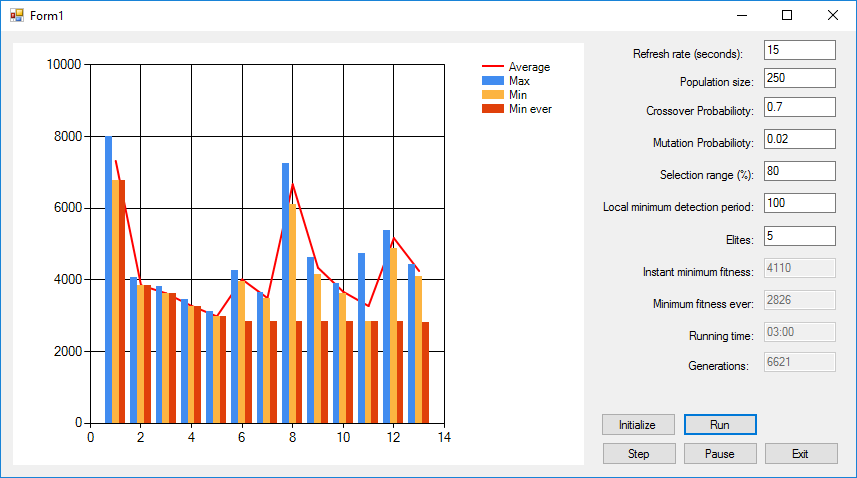
באמת רוב השיפור הוא בדקה הראשונה, אך גם אחרי הדקה הראשונה ממשיך להיות שיפור אומנם קטן יותר.

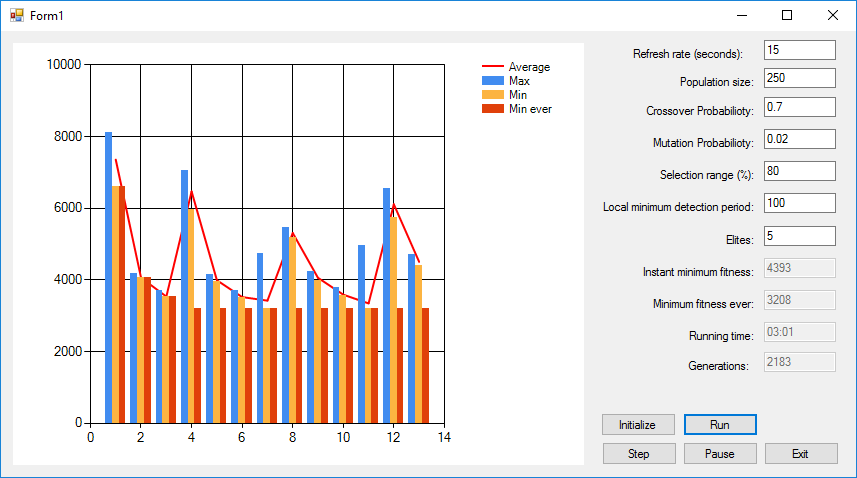
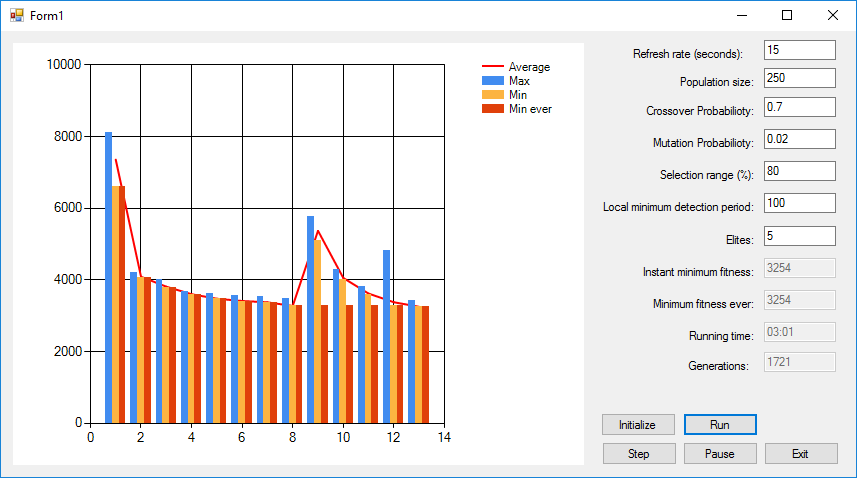
סעיף 3

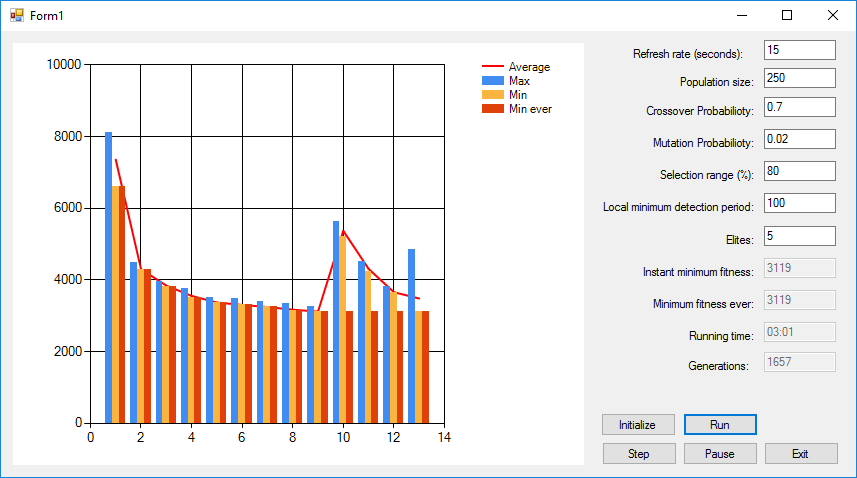
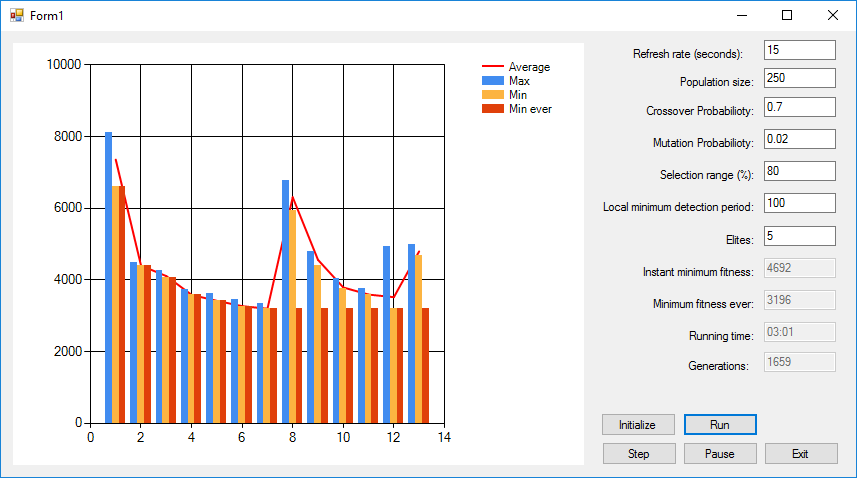
להלן טבלה של התוצאות אחרי 3 דקות, בהמשך צרפתי את הגרפים.

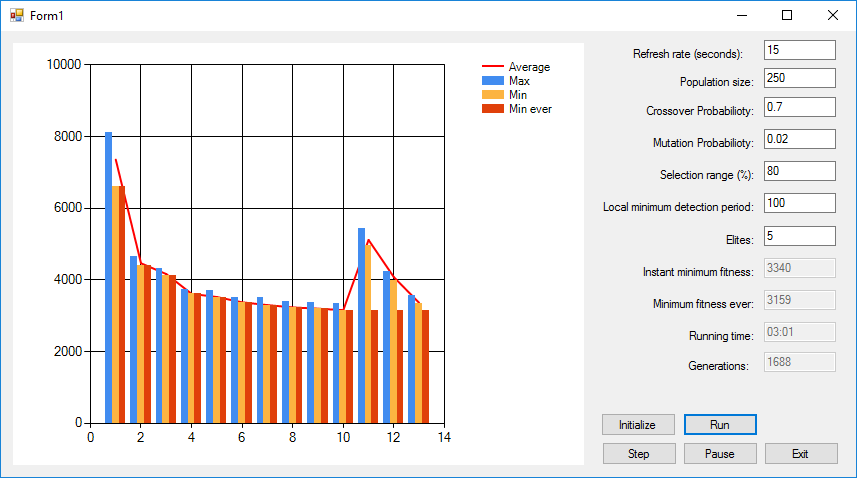
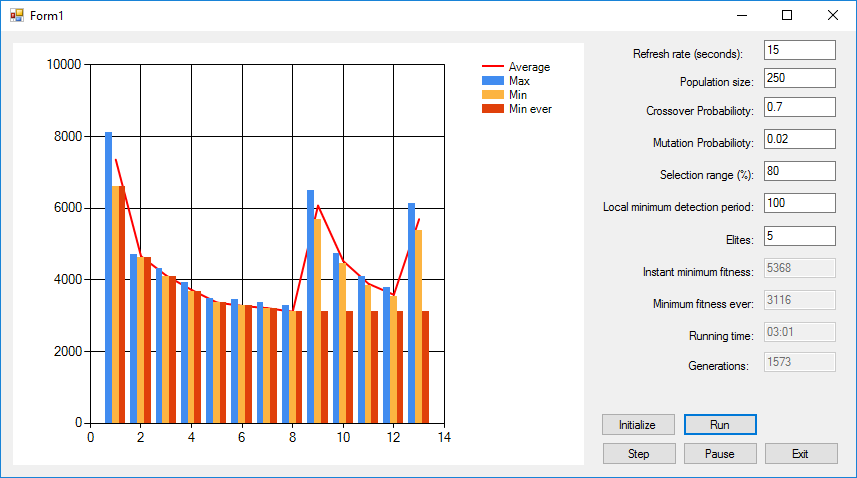
המרחק הממוצע מהרצוי מצוי מחושב ע"י המינימום לחלק ל 50 זוגות, ל 3 פריטים, שני מרחקים. כלומר המינימום חלקי 300.

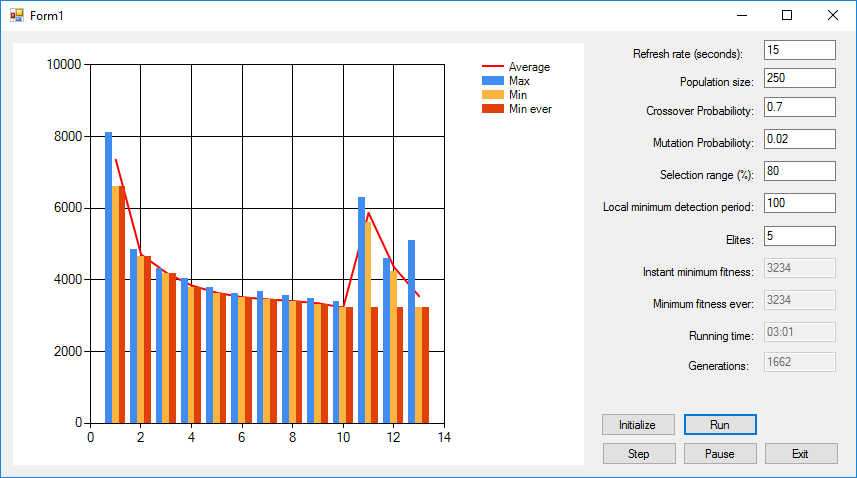
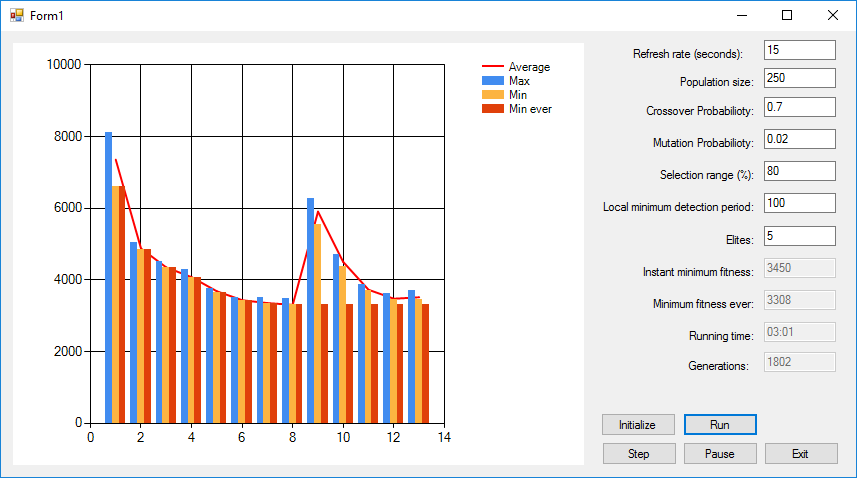
|  |  |  |
| --- | --- | --- |
| מספר ריצה | מינימום באוכלוסיה | מרחק ממוצע מהרצוי למצוי |
| 1 | 3208 | 10.693 |
| 2 | 3254 | 10.846 |
| 3 | 3119 | 10.396 |
| 4 | 3196 | 10.653 |
| 5 | 3159 | 10.53 |
| 6 | 3116 | 10.386 |
| 7 | 3234 | 10.78 |
| 8 | 3308 | 11.02 |
| 9 | 3240 | 10.8 |
| 10 | 3165 | 10.55 |

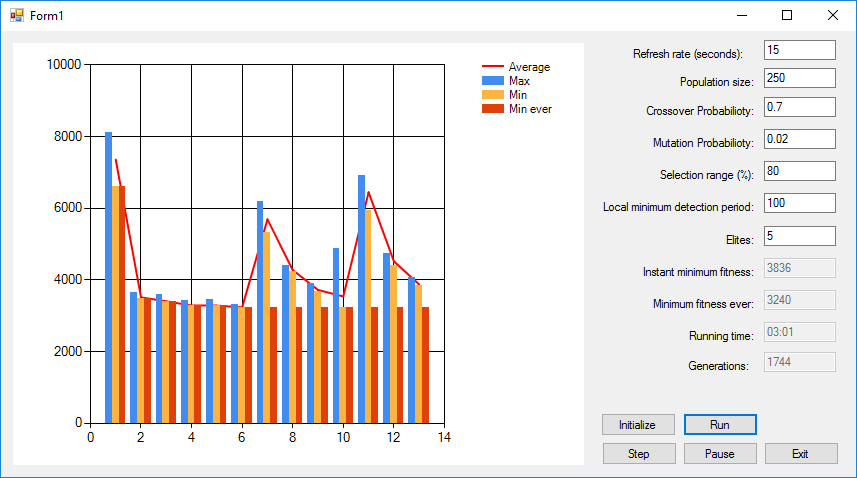
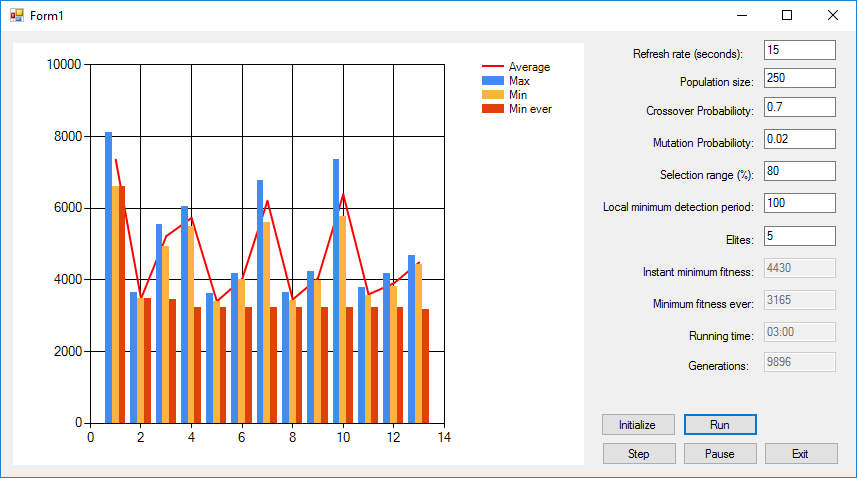
המרחק הממוצע של כל פריט (אישה,גבר או כלב) מהרצוי למצוי של כל הריצות הוא 10.665 מתוך 50, עם שונות נמוכה. שזה טוב לדעתי, אני הייתי קונה את זה מיד :). כלומר, האלגוריתם מגיע לתוצאה יציבה.

סעיף 4

פתרון בודד הוא רשימה של 50 שלישיות, שבכל שלישייה יש גבר אישה וכלב והשלישייה מתארת גם את המיזוג של שלושתם. כלומר, הגבר שודך עם האישה והכלב מאותה השלישייה וכ"ו.

בשחלוף אני מגריל מספר בין 0 ל 49 וחותך את הכרומוזום במיקום הזה. לוקח את החלק הראשון מההורה הראשון ואת ההמשך מההורה השני. אחרי זה אני בודק עם לא נוצרו שגיאות, כלומר לדוגמה איזה שמופיעה בשני שלשלות. אם כן אני מחליף אחת מהמופעים שלה עם אישה שלא נמצאת באף שלישייה.

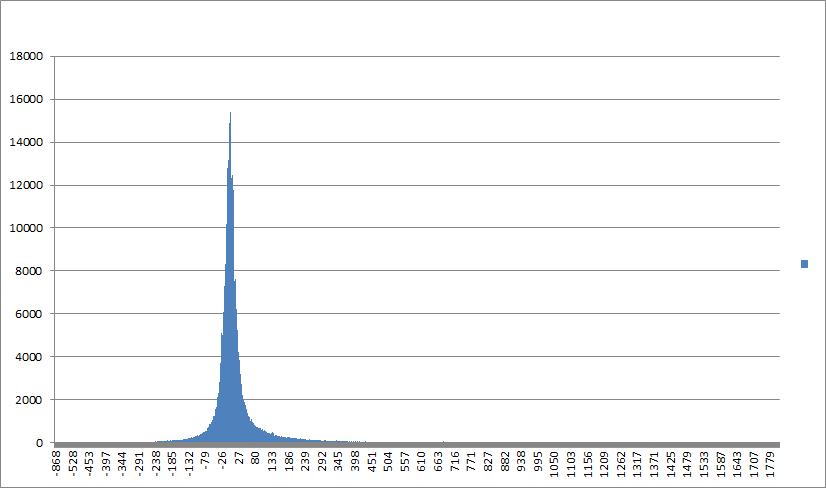
במוטציה אני מגריל זן. כלומר, גבר אישה או כלב. נומר שנבחרה איזה. אני מגריך מספר בין 0 ל 49 ומחליף את האישה בשלישייה הראשונה עם האישה בשלישייה השנייה.

סעיף 5

השחלוף יוצר בהסתברות שווה צאצאים טובים יותר וטובים פחות.

בדקתי את זה באמצעות היסטוגרמה של הפרשי הקשירות בין הצאצא אחרי שחלוף לממוצע הקשירות של הורים.

בגלל שמימשתי בעיית אופטימיזציה, כלומר ערך קשירות נמוך יותר הוא טוב יותר, הפרש שלילי הוא שיפור וחיובי הוא גריעה. וזו ההיסטוגרמה:



כלומר, הערכים באופן שווה סביב האפס

סעיף 6

בממשק יש פרמטר בשם 'local minimum detection period' והוא מבוטא במספר דורות.

אם אחרי מספר דורות x אין פישור באוכלוסייה כלומר, ערך הקשירות המינימאלי לא יורד. אם מזהה מינימום לוקלי ונכנס לתקופה/פזה של ניסיון להתחמק מהמינימום הזה. הניסיון הוא באמצעות:

1. במידה ומינימום לוקלי זוהה
   1. שומר את 5 הפריטים הטובים ביותר באוכלוסייה בצד.
   2. מעלה את Pm כפול 4, עד תקרה של 1
   3. מחליף את כל האוכלוסייה לאוכלוסייה חדשה, באופן רנדומלי.
2. נותן לאוכלוסייה החדשה להתפתח במשך x כפול 4 דורות
3. אחרי x כפול 4 דורות:
   1. מחזיר את 5 הפריטים ששמרתי בצד בחזרה לאוכלוסייה במקום 5 הגרועים ביותר.
   2. מחזיר את הערך של Pm