1. קובץ הREADME לתרגיל 3

2. מגישים:

שמעון בריזל, 214552143

שלמה זלמן קאהן, 327517066

3. הסבר כללי של התרגיל:

יצרנו מ0 פרויקט המסוגל לבצע מניפולציות ברמת הפיקסל והתמונה: השוואות, פעולות לוגיות, שינויי קנה מידה וטרנספורמציות.

4. רשימה של הקבצים שיצרנו, עם הסבר קצר לגבי תפקיד הקובץ:

Pixel – ייצוג פיקסל בודד עם פעולות לוגיות והשוואה

ImageDataStructure - ניהול מטריצת פיקסלים: הקצאה דינמית, גישה ושינוי פיקסלים

Image - ניהול תמונות: העתקה, מניפולציות לוגיות ואריתמטיות, שרשור וכו.

Main - בדיקות לכל תכונות הספרייה: פעולות על פיקסלים, תמונות, גישה לפיקסלים, ושינויי קנה מידה

5 .מבני נתונים עיקריים ותפקידיהם:

**ImageDataStructure** - מטריצת פיקסלים: מנהלת את כל הפיקסלים בתמונה, מאפשרת גישה ושינוי של פיקסלים בודדים.

6. אלגוריתמים הראויים לציון:

7. תיכון:

הפרויקט בנוי בצורה מודולרית, כאשר כל מחלקה ממלאת תפקיד ייחודי במערכת, תוך הפרדה ברורה בין האחריות של כל רכיב:

1. **Pixel**:  
   מייצג יחידת פיקסל בודדת. תומך בפעולות לוגיות פשוטות. זה מאפשר לבצע מניפולציות פרטניות על פיקסלים תוך שמירה על פשטות הקוד.
2. **ImageDataStructure**:  
   משמש כמחלקת ניהול למטריצת הפיקסלים של התמונה. המחלקה מבצעת הקצאה דינמית של זיכרון, גישה לפיקסלים לפי מיקום (באמצעות אופרטור ()), ושינויים בפיקסלים. מחלקה זו מתפקדת כבסיס שעליו מתבססת מחלקת ה-Image לעבודה עם הפיקסלים.
3. **Image**:  
   מייצגת תמונה שלמה ומאפשרת ביצוע מניפולציות ברמת התמונה בעזרת אופרטורים מוגדרים.

**מבנה התקשורת בין המחלקות**:

* + **Pixel** משמש כאבני הבניין, עם פונקציות פשוטות לטיפול בצבע.
  + **ImageDataStructure** מקשרת בין פיקסלים ומספקת גישה יעילה למטריצת הפיקסלים.
  + **Image** עושה שימוש במחלקת ImageDataStructure לניהול המטריצה, תוך הרחבת הפונקציונליות לתפעול תמונה שלמה.

8. באגים ידועים:

9. הערות אחרות: