## מטלה 0 – מבוא לתכנות מובנה עצמים

מגישים:

נתנאל חוגי, ת"ז: 203553490

שמעון חג'ג', ת"ז: 311367536

חננאל מירון, ת"ז: 302326137

עבור המטלה השתמשנו באפליקציה Wiggle-Wifi, שאוספת ושומרת נתונים של רשתות אינטרנט קרובות. את הנתונים ניתן לייצא לקבצי Csv וקבצי



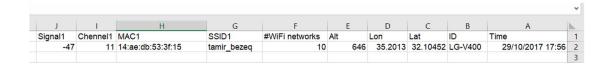
צילום מסך מתוך האפליקצייה.

האפליקציה יצרה עבורנו קובץ CSV שמכיל את הנתונים הבאים: כתובת MAC, שם רשת, מיקום, תדר, עוצמת קליטה וכ'ו.

בנוסף בקובץ מופיעים נתונים עבור המכשיר בו התבצעה הסריקה.

בחלק הראשון של המטלה, לקחנו את הקובץ שיוצא מהאפליקציה וכתבנו עבורו קוד ג'אווה(-wifi) בחלק הראשון של המטלה, לקחנו את הפרטים ומעביר אותם למבנה נתונים(מערך דו-מימדי), ומדפיס אותם לקובץ CSV חדש לפי הפורמט המבוקש במטלה.

הקוד מקבל נתיב של תיקייה עם קבצי CSV מהאפליקציה, עובר על כולם ושומר למטריצה לכל היותר 10 רשתות מכל דגימה(שומר את הרשתות בעלות עוצמת החיבור הגבוהה).



לאחר ההרצה קיבלנו את הקובץ הנ"ל,

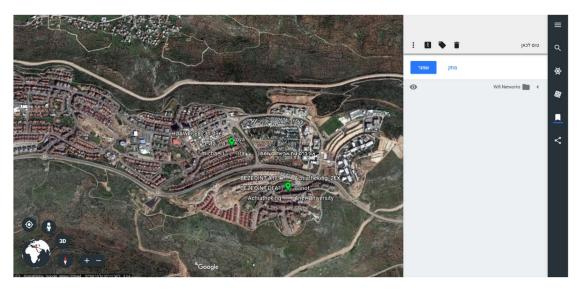
שעת הדגימה נלקחה משם הקובץ מהאפליקציה, מזהה מכשיר מופיע באותו קובץ כ"model", נתוני מיקום נלקחו מהרשת הראשונה בכל דגימה. במהלך המעבר על התיקייה, הקוד למעשה מדלג על קבצים בעלי סיומת שונה מcsv, כמו כן גם על קבצים עם סיומת תקינה, אך ריקים/ללא רשתות.

עבור החלק השני והשלישי של המטלה, כתבנו את הקוד CSV TO KML.

הקוד קורא את הקובץ שיצרנו בחלק הראשון, מסנן את הנתונים לפי מה שהגדרנו ומעביר את הנתונים לקובץ KML.

הקובץ KML נפתח בgoogle earth, ומציג את שמות הרשתות ונתונים עליהם על גבי מפה, לפי הקורדינטות של הדגימה.

הקוד מדפיס לאחר הסינונים מבנה של קובץ KML עם נתוני הרשתות והמיקומים.



צילום מסך מגוגל ארת.

את בדיקת הקודים שכתבנו, ביצענו על הקבצים מהאפליקציה ועל הקבצים שהועלו למודל.

## <u>רשימת הפונקציות(wifiScanner):</u>

```
public static void filesReader(String folderPath) throws Exception{
```

הפונקצייה מקבלת מחרוזת(שם תיקייה), עוברת על הקבצים, מעבירה את הנתונים מהקבצים המתאימים אל מערך דו מימדי ולבסוף מדפיסה את כל הנתונים לקובץ CSV לפי הפורמט הדרוש במטלה.

```
public static int findMinRow(String[][] arr){
```

הפונקצייה מקבלת מטריצה, ומחזירה את מספר השורה במטריצה עם הנתונים של הרשת בעלת עוצמת הקליטה הנמוכה ביותר.

```
public static int findRssi(String s){
```

הפונקצייה מקבלת שורה עם נתוני רשת מהקובץ שיוצא מהאפליקציה, ומחזירה את עוצמת הקליטה.

```
public static String timeAndExt(File file){
```

הפונקציה מקבלת קובץ, ובודקת לפי שם הקובץ את תאריך הדגימה, ובודקת האם הסיומת תקינה. הפונקצייה מחזירה את זמן הדגימה, או "ERROR" במידה והקובץ לא תואם.

```
public static int wifiNumber(File file) throws IOException{
                                  הפונקצייה מקבלת קובץ ומחזירה את מספר הרשתות שנמצאות בו.
public static String userName(String s){
                                                      פונקצייה המחזירה את מזהה המשתמש.
 public static String[] findCoordinate(String s){
                                                   פונקצייה המחזירה את הנתונים הגיאוגרפיים.
                                                        רשימת הפונקציות(CsvToKml):
private static String[][] ReadFile(String name) throws IOException{
                   פונקצייה המקבלת את הקובץ שיצרנו בחלק הראשון, ומעבירה את הנתונים למטריצה.
public static void UnFilter(String fileName) throws IOException{
         הפונקצייה מקבלת שם קובץ, ∪מעבירה את הנתונים(ללא סינון) לקובץ KML(בעזרת פונקציות עזר).
*public static void timeFilter(String fileName, int fromHour, int toHour) throws IOException{
   פונקציות סינון לפי שעה(מקבלת שעה מוקדמת ושעה מאוחרת) ומציגה רק את הבדיקות שנעשו בין השעות
                                                                             הללו(כולל).
public static void dateFilter(String fileName, int day, int month, int year) throws IOException{
                                                                    סינון לפי תאריך מדוייק.
```

public static int rowNumber(String file) throws IOException{
פונקציית עזר שבודקת כמה שורות(רשתות) יש בקובץ.

public static void IDFilter(String fileName, String ID) throws IOException{
סינון לפי מזהה מכשיר(מדוייק).

private static void toCsv(String [][] arr,String newFile) throws FileNotFoundException{

 $\textbf{private static void } \textbf{toKml} (\texttt{String [][] arr, String newFile}) \textbf{ throws } \textbf{FileNotFoundException} \{ \textbf{static void } \textbf{toKml} (\texttt{String [][] arr, String newFile}) \} \}$ 

פונקצייה היוצרת קובץ KML עם הנתונים. הפונקציה למעשה מדפיסה את מבנה הקובץ בhtml, ומכניסה את הנתונים הרלוונטיים מתוך המטריצה למיקום שלהם בקובץ.

## <u>באגים:</u>

פונקצייה היוצרת קובץ CSV חדש, אחרי סינון.

1. התוכנה יוצרת קובץ KML עם נקודות לפי מיקומי הבדיקות, בכל נקודה מופיעים כל הרשתות(10 לכל היותר) מכל דגימה. הרשתות לא מוצגות לפי "המיקום החזק" שלהן.

2. חסר פילטר המסנן את הנתונים לפי מיקום.